

この資料は日本エム・テクノロジー学会員専用です。
この資料を学会員以外がコピーしたり、学会員以外に配布することを禁じます。

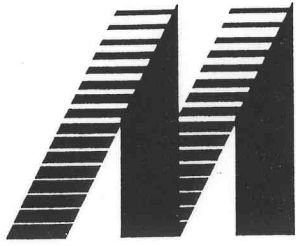
Copy right : M Technology Association - Japan

日本エム・テクノロジー学会事務局
〒259-1193 神奈川県伊勢原市望星台
東海大学医学部・基礎医学系
大櫛陽一

Tel: 0463-93-1121 ext. 2140

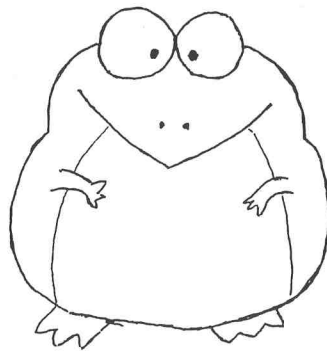
Fax: 0463-96-4301

Email: youichi@keyaki.cc.u-tokai.ac.jp



*Technology
Association
Japan*

第21回
日本エム・テクノロジー学会大会
予稿集



199⁴~~3~~年8月5日(金)~8月7日(日)
茨城県 つくば市

第20回日本エム・テクノロジー学会大会予稿集

目次

学会大会案内	1
会場案内図	2
プログラム	3
特別報告、特別講演、海外報告、プロダクト紹介、ワークショップ	7
一般演題	31

全大会第一卷

第21回日本エム・テクノロジー学会大会

日程 1994年 8月5日～8月7日
会場 筑波大学医学専門学群 臨床講義室 B
大会長 岡田 好一 (筑波大学)
大会副会長 木村 一元 (独協医科大学)

プログラム委員会

委員長 木村 一元 (独協医科大学)
委員 今泉 幸雄 (サンド薬品株式会社)
委員 嶋 芳成 (日本ダイナシステム株式会社)
委員 田久 浩志 (東邦大学)
委員 本多 正幸 (千葉大学)

M言語フェア実行委員会

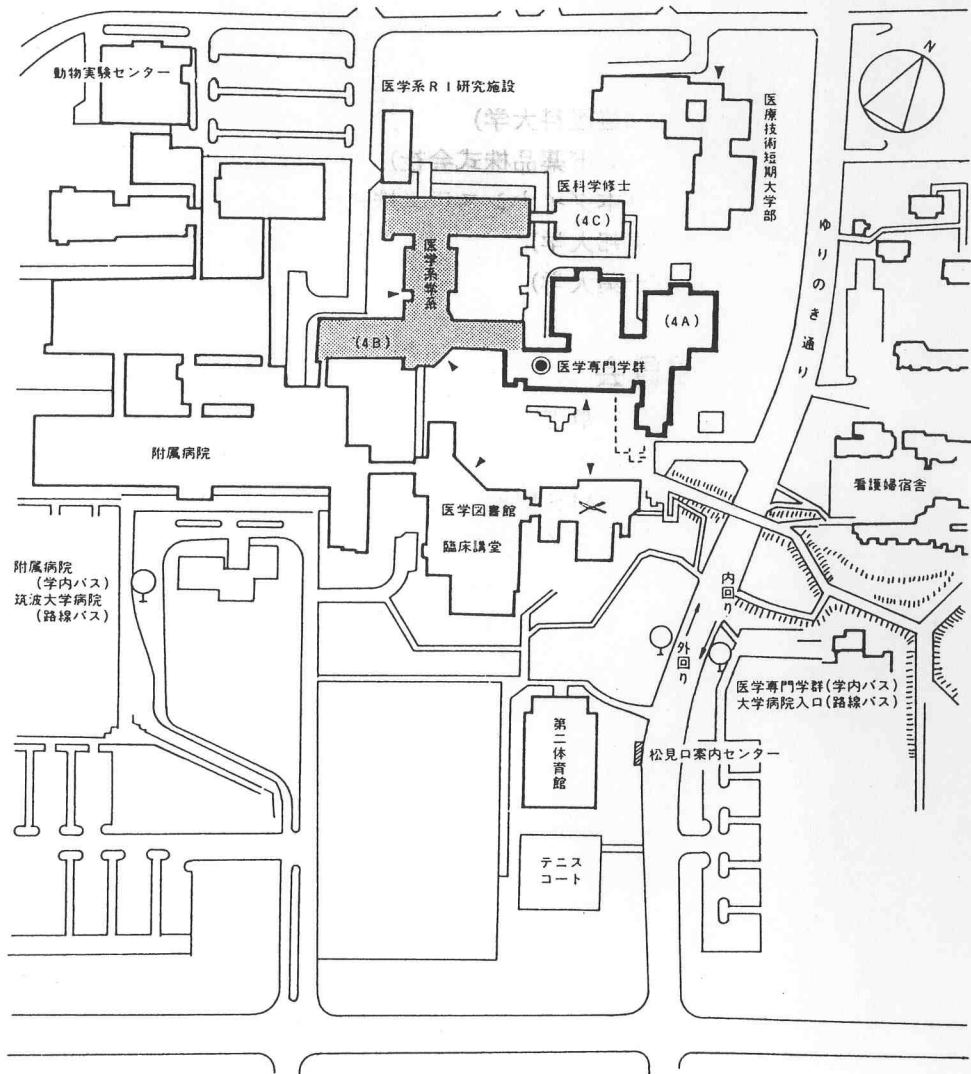
委員長 小倉 勉 (住友電気システムズ株式会社)
委員 今井 敏雄 (日本デジタルイクイップメント株式会社)
委員 松本 重雄 (日本MSM株式会社)
委員 山本 純史 (日本ダイナシステム株式会社)

協賛

住友電気システムズ株式会社
日本ダイナシステム株式会社
日本デジタルイクイップメント株式会社
株式会社 高崎共同計算センター

筑波大学医学専門学群周辺施設配置図

会場：臨床講義室 B (医学図書館棟 3階)



凡 例



学群棟



学系棟



通常の出入口



医学事務区事務室



バス停



食堂

第21回日本エム・テクノロジー学会大会プログラム

8月5日(金)

- ◎ M言語初級講習会 【13:00-15:00】
日本エム・テクノロジー学会
- ◎ 特別講演(1) 【15:00-15:30】
司会 岡田 好一 (筑波大)
高田 彰 (筑波大)
筑波大学附属病院における病歴資料・情報の共有化と情報システム
- ◎ ワークショップ 【15:30-18:30】
今泉 幸雄 (サンド薬品)・岡田 好一 (筑波大)：進行役
オブジェクト指向とM言語
- 評議員会 【18:30-19:00】
場所：医学食堂・会議室

8月6日(土) (1)

◎ 一般講演(1) 大学病院 【9:00-10:00】

座長 田久 浩志 (東邦大)

鈴木 隆弘 (千葉大) 他

千葉大学医学部附属病院に於ける、病床管理システムの導入について

山下 芳範 (福井医大)

福井医科大学新総合病院情報システムにおけるM言語の利用と概要

笹川 紀夫 (島根医大) 他

島根医科大学教育研究システムの概要とシステム運用について

870 土田 伸 (西川町役場) 他

西川町総合保健医療福祉サービスにおける地域保健・医療・
福祉行政総合支援システムの構築

199,000人

◎ 一般講演(2) 新世代M言語 【10:00-11:00】

座長 鈴木 利明 (日本ダイナシステム)

今井 敏雄 (日本DEC)

Mとクライアント / サーバーコンピューティング

38-41

前海 好昭 (日本DEC)

Open環境で動作する新世代DSM

42-43

今泉 幸雄 (サンド薬品)

EsiObjectsの概要

19-29

◎ 海外報告(1) 【11:00-11:30】

司会 木村 一元 (独協医大)

山下 芳範 (福井医大)

北米MTA報告

MDC報告 -MDC update-

◎ 特別講演(2) 【11:30-12:00】

司会 山本 和子 (島根医大)

里村 洋一 (千葉大)

M言語によるクライアント・サーバーシステム構築

8月6日(土) (2)

- 総会 【13:00-14:00】
 司会 河村 徹郎 (鈴鹿医療科学技術大)
- ◎ 特別報告 【14:00-15:00】
 大櫛 陽一 (東海大) : 世話人
 JIS原案作成について 8-8
- ◎ 特別講演(3) 【15:00-16:00】
 司会 河村 徹郎 (鈴鹿医療科学技術大)
 若井 一朗 (マンプス研)
 M言語JIS化のインパクト
- ◎ 招待講演 【16:00-16:30】
 司会 里村 洋一 (千葉大)
 J.Dumas (Veterans Administration)
 The USA MUMPS Activities
- ◎ 海外報告(2) 【16:30-17:00】
 司会 本多 正幸 (千葉大)
 M. アマラル (千葉大)
 M Technology in Brazil: Status Report
- ◎ プロダクト紹介 【17:00-18:00】
 J. Jhung (OpenTech) *tel 82-2-555-611x msmasia@*
Fax 82-2-56x-511x
 E. Pape (InterSystems)
 DTM Product Directions
 B. Mappes (Micronetics)
 MUMPS Market Assessment and Opportunity
- 懇親会 【18:20-19:00】
 場所 : 医学食堂

8月7日(日)

-
- ◎ 特別企画勉強会(M言語システム中級)
- | | |
|--------------------|---------------|
| 住友電気システムズ株式会社 | 【 9:00-10:00】 |
| 日本デジタルイクイップメント株式会社 | 【10:00-11:00】 |
| 日本ダイナシステム株式会社 | 【11:00-12:00】 |
-
- ◎ 一般講演(3) Mツール(1) 【13:10-14:20】
- 座長 笹川 紀夫 (島根医大)
- 馬場 謙介 (国立埼玉病院)
異巾文字列処理の考察
- 田久 浩志 (東邦大)
GUIツール、MGRの概要と紹介
- 馬場 謙介 (国立埼玉病院)
文字データの検索に関する考察
- 大櫛 陽一 (東海大) 他
新開発環境Easy-Toolの開発
- ◎ 一般講演(4) Mツール(2) 【14:30-15:40】
- 座長 藤江 昭 (住友電気システムズ)
- 馬場 謙介 (国立埼玉病院)
経皮的冠動脈拡張術(PTCA)施行症例データベースに使用した
データ等の読込に関するSmall Concepts
入院病歴管理システム・経皮的冠動脈拡張術症例データベースに
応用した呼出機能を持つ編集可能疑似READ命令
- 波川 智亮 (日本ダイナシステム) 他
MS-WindowsとDTM環境におけるGUI開発方法の比較
- 佐藤 比呂志 (日本DEC)
M環境と外部世界との融合の試み
- 木村 一元 (獨協医大) 他
獨協医科大学病院薬剤部 薬歴管理支援システム
-
- 54-57

特別報告

招待講演

特別講演

海外報告

プロダクト紹介

ワークショップ

特別報告：J I S 原案作成について

大櫛陽一、本多正幸、藤江昭、若井一朗、高橋隆、里村洋一、山本和子、
河村徹郎、木村一元、山下芳範、嶋芳成、小林泰道、松本重雄、永井肇、
岩田稔、根岸重夫、丸尾誠史、加山英男、内山誠作

日本電子工業振興協会MUMPS言語原案作成委員会

1. 昨年度の作業結果

1993年4月より、J I S 原案を作成するために、“ISO/IEC 11756 First edition 1992-12-15 INTERNATIONAL STANDARD Information technology- Programming languages - MUMPS”の翻訳作業を開始した。この作業のために、一太郎V4用の文部省用語辞書、JIS X 0001~0022用語英和对訳表、MUMPS用語英和对訳表を作成し、作業の正確化と効率化を計った。委員会の全体会議は、4回開かれた。全体会議以外に、幹事会が約5回開かれた。翻訳の正確化、言葉の統一、J I S 文書形式の遵守、日本語処理の追加、解説文書の追加、英和对訳の追加、生成規則一覧（附属書B）と索引の日本語順ソート化などと、思っていたより大変な作業が続いた。何度も改訂を繰り返し、日本工業標準調査会情報部会企画調整専門委員会委員の早稲田大学寛捷彦教授の査読と、日本電子工業振興協会言語標準化調査委員会での検討をいただいた。この結果、日本工業規格[案]中間報告として、“日本工業規格[案] プログラム言語MUMPS J I S X 30xx-19xx
1994年3月 社団法人日本電子振興協会”が完成した。

2. 今年度の作業予定

すでに、全体委員会が1回、幹事会が1回開かれた。これらの会において、中間報告書の最終的チェックと、原案としての体裁の見直し、J I S 印刷時の注意点のまとめなどが行われた。現在の原案は12版となっており、6月末に通産省工業技術院に提出されることになっている。7月からは、MUMPS独自の用語についての日本語訳を、J I Sの用語委員会（INSTAC）に用語登録申請を行い、審議をお願いすることとなっている。委員会は、本年度に、残り2回が予定されている。ここでは、J I S化の進捗に合わせた検討、次のバージョンの動きについての情報交換、次期ISOのJ I S化の検討などを予定している。

M 言語 JIS 化のインパクト

マンブスシステム研究所

若井一郎

全てのコンピュータ言語にはライフサイクルがありそれが一国の工業規格に制定される以前に死語となるものが多い。工業規格となった場合には、その規格で生産されたものを後から否定することができぬので一朝一夕に死ぬことはない。然し、ユーザがいなくなればネジの規格のように百年生き延びても、産業界の供給が止まり、博物館入りとなる。日本はM言語のJIS化に拒否票を投じた国でありながら肯定票を投じた米国以外のSC22国では最初に日本工業規格を制定する不思議な国である。その上、次期ISO制定(1995?)からはJIS90というX0201とX0208による「日本文字セット」を附属書A(規定)として国際規格とし、日本に輸出するM言語装備はこれを守らねばならないことになる。

国際規格化と同時に「日本文字セット」が規定としてISO規格に入ったことは、他のJIS規格言語に比して日本語が早期にM言語の早期化を意味して、常識では国内に装備者が少ない所でJIS化が進むとは考えられない点で、その歴史的な背景にはユーザの力による所が大きかったと言わねばならない。特筆すべきものは昭和54年3月18日に第1回「標準カナMUMPS委員会」(大楠陽一委員長)が発足しANK1バイト文字に関するM規格を作り始め、この成果がMDCに報告され、ASCII文字以外の文字セットの問題に注意を促してきたこと、やがて昭和59年11月24日から新たに2バイト文字の問題も含み「日本語マンブス標準化委員会」に止揚された時から8回の会合で昭和61年1月16日に全てのベンダーの意見一致の下に「イ草案」が公表され、これらが全てMDCC-Jの名でMDCに反映されて来たこと、これをヨーロッパが応援し、MDCで国際化をテーマとするSC12が英国のDiamond氏の下にできた、などの経緯である。果たして、M言語のISO化に肯定票を投じたがMTAの活動のない中国では中国工業規格としてM言語を取り上げるとき大きな混乱と紆余曲折が予想される。韓国でもユーザ間に共通の認識があるとは言えず、工業規格化の道程は長いと思われる。

M言語JIS化の主導者がユーザであった以上、M言語JIS化のインパクトはユーザ活動の与えるインパクトを抜いて考える訳にはゆかない。ユーザ活動としてのM言語JIS化のインパクトたるべきものを以下に挙げ、放置してもインパクトが生じるかどうかを考えて、冷厳に期待度を検討したい。

- アジア2バイト文字諸国の問題を解決する活動がM言語の販路を広げる。
- Unicode採用による国際化—Unicode Consortiumは実現性があるか(日中韓へのメリット?)
(アメリカソフトハウス・日本大手コンピュータ関連会社が参加の市場拡張型
Unicode(Windows NT) <=> (Chicago) Shift-JIS)。
- 日本語マンブス標準化委員会を活性化する。
- 国家規格を支える産業の積極的装備を刺激する。
(COBOL, FORTRAN, C, SQLグループとの協調)
- SC22を取り込む。->MDCへの発言を強化。
- ISO化の過程で日本SC22の発言を強化する。MのISOアリーナへの発言を強化する。
- ユーザ人口の維持と拡大に通じる標準メディア(教育資源)を確立する。日本MTA学会。
- M技術とコンピュータ大衆との接点を広くする。
- Mの認識と受容へのストラテジックプランニングが不可欠になる。
- 市場経済原理を離れた所で進化は求めない。

M言語によるクライアント・サーバーシステム構築

千葉大学医学部附属病院 医療情報部 里村 洋一

1. ネットワークとMUMPS

もともと分散処理とスタンドアロンシステムのために開発されたM言語であるが、ネットワークがもてはやされ、応用システムの規模が大きくなるにつれて、心ならずもネットワーク対応を迫られるようになったといえる。DDPと呼ばれる機能はこうしたネットワーク上でのM言語の動作を保証するものとして開発され、多くの応用システムで活躍してきた。しかし、グローバル変数アクセスという、本来システムバスに直結したルートで行われるべきプロセスが、いかに回線の容量が増加したとはいえ、イーサネットなどの外部回線を経由して行われるとなれば、そのトラフィックをこなすのは至難のことである。特に検索のために膨大な数のノードをたどるような場合、ルーチンの実行速度はほとんど回線の効率に支配される事になってしまう。

この問題を解決しないかぎり、クライアント・サーバー方式のシステム構築はM言語のもとでは困難である。そこで、ユーザー側として、いくつかの解決方法を考察してみた。

2. DDPの効率

われわれはSUN社のワークステーション(S-4/470)をサーバーとし、同じくSUNのS-4/370をクライアントとして、これらをイーサネットでつなぎ、U-MUMPSとそのDDPによって、オーダーエントリーと電子カルテのシステムを運用している。現システムで実際的な運用上の深刻な問題は発生していないが、システム稼働のピーク時間帯や、バッチ的なプログラムによって大量のデータ検索が発生すると、明らかなレスポンスの遅れがみられる。

イーサネット上のトラフィックは最大能力の2%程度に限られていることから、回線容量の問題ではないと推察し、DDPの処理能力について実験してみた。

ほとんどのアプリケーションが止まっている夜間において、2つのノードを持つ同一のグローバルをクライアント側からDDPを介して検索した場合と、サーバー内で検索した場合を比較した。グローバル50万件中から1000件をランダムにアクセスすると、サーバー内では約30秒、クライアント側からは平均約43秒を要した。1000件の連続性のあるアクセスでは、サーバーで2秒、DDP経由で22秒であった、サーバーでのDDPとアプリケーションの処理に優先度の差が与えられていない状態では、この差13-20秒はDDPのプロセスによって生まれたと解釈できる。

この13秒の意味するところは大きい。通常、サーバーにおけるグローバルアクセスはディスクキャッシュによって高速化されており、キャッシュに存在するデータへのアクセス時間は、無視できる程度に小さい(1000件程度は1秒以内)が、DDP経由のアクセスでは、DDPの処理時間を短縮する方法がない。従って、ここで計測された1000件あたり13秒は、DDP経由グローバルアクセスの全てにかかる時間負荷である。M言語の得意とする単一のグローバル

データへアクセスは大きな問題とならないが、集計や選択のために多数のノードを検索する場合には、この負荷は堪え難い。サーバーマシンの能力が高まりサーバー内でのデータ検索が高速に遂行できる時代となってきたが、このままでは、M言語の世界でその高性能を生かせない。

3. QUERY言語による解決

M言語のグローバルデータ処理プロセスにSQLの機能を組み込むのが、標準的な解決方法である。この場合、従来のグローバル記述とSQL記述の双方がアプリケーションで選択出来れば、効率のよいシステムが期待出来る。B-T r e eデータベースに対するSQLを実現するのはそれほど困難とは思えないので、ベンダーに期待するところ大である。

システムレベルでの実現がさしあたり難しいとすれば、M言語によって簡易クエリー言語の機能を作成するのも一法であろう。アプリケーションの形を限定すれば、ユーザーの手で開発できる。パラメータパッシング機能を用いて、JOB命令によってサーバーの汎用検索プログラムにリモートジョブ起動をかけ、結果をクライアントのグローバル変数に受け取る。汎用検索プログラムはパラメータから合成する間接実行文で構成することで柔軟性を持たせることが可能である。

4. アプリケーションによる解決

アプリケーションレベルで解決する方法には、様々のものが考えられるが、ここでは、病院情報システムにおいて実用的と思われる方法を二つ挙げてみる。

4. 1 アプリケーションを2分割する方法

DDPに隘路があるとすれば、DDPの利用をできるだけ節約するのが、単純な解決方法である。たとえば、サーバーに存在する多くのグローバルデータから、ある条件に適合するデータだけが必要であるような場合、クライアント側でグローバル記述に拡張シンタックスをつけてDDP経由で検索するのが通常の方法である。これでは、すべてのグローバル記述に対応してDDPを介してデータ伝送が1往復ずつ発生するわけで、DDPに対する負荷は対象グローバルのサイズに比例する。これを避けるには、サーバーにおける検索プログラムとクライアントにおける検索結果の利用プログラムを別々のものとするのがよからう。

こうすれば、DDPの負荷はクライアントからサーバーに対するリモートジョブ起動と条件に適合するデータのサーバーからクライアントへ向けての1方向伝送のみとなり、画期的に負荷が減少する。その効果は、検索対象グローバルのサイズが大きければ大きいだけ、条件適合の比率が少なければ少ないだけ大きい。

その効果を実験で確かめてみた。

<実験条件>

サーバー上に記録されている1000件の患者ファイルを対象にして、ある条件に適合する患者を選んで、(この実験では、最終来診日が特定の日付より前であるとの条件をつけた：この条件

を満たす患者の数はチェック日付の与えかたによって表1のように変わる) その患者番号をクライアントのローカル変数にセットした。

データアクセスの初回はグローバルがキャッシュ上に無いため、1000件につき4秒ほどのディスクアクセスのためのオーバーヘッドが存在する。その影響を避けるため、同一データを反復検索し、サーバーのグローバルキャッシュ上にすべてのデータがそろっている状態で比較した。

<実験結果>

表1. 単一プログラムとクライアントとサーバーでのプログラムを別とした場合の処理時間

ヒット率	クライアントの単一プログラム	プログラム2分割
600/1000	27秒	6秒
395/1000	26秒	4.5秒
240/1000	26秒	3秒
135/1000	27秒	2秒
71/1000	25秒	2秒

ヒット率が10%以下の場合には当然、60%の場合でさえも、処理時間は1/5程度になることが証明された。

この方法は、プログラム開発がやや複雑となるが、スループットの向上は明らかである。ただし、その効果は、条件検索のときにのみ現れる。

4. 2 テンポラリーデータベースを作る方法

病院情報システムのアプリケーションの多くは、上記のような条件検索は少なく、ほとんどが、特定の単一グローバル参照である。そのようなアプリケーションでは、単位の大きなデータベースに直接アクセスするより、特定の業務に必要なデータのみを集めた小規模なデータベースをあらかじめ臨時に作成しておき、その範囲でデータ処理して、最終的な結果のみを元のデータベースに戻すという手法が考えられる。具体的な応用としては、外来診療のために、当日来診予定の患者のデータを、センターのデータベースから抽出して、あらかじめユーザーの使用するWSやパソコンなどに集めておく事がある。すなわち、臨時の分散データベースを図るのである。その患者を診療するWSやパソコンが特定できる場合は、前日夜間の回線が暇な時間帯を選んで一括転送すればよからう。そうでない場合は、あらかじめ、転送用のファイルを作成しておき、患者の診療が始まったときに、そのWSやパソコンへ転送することで、目的が達せられる。

具体的な、応用の例として図1にデータの流れを示した。

ここでは、クライアントとしてパソコン (M言語でアプリケーションが乗っている) を想定している。データ転送には、DDPを使わずUNIXの rcp コマンドを使う、UNIXファイル単位の転送ならば、効率は数段優れている。

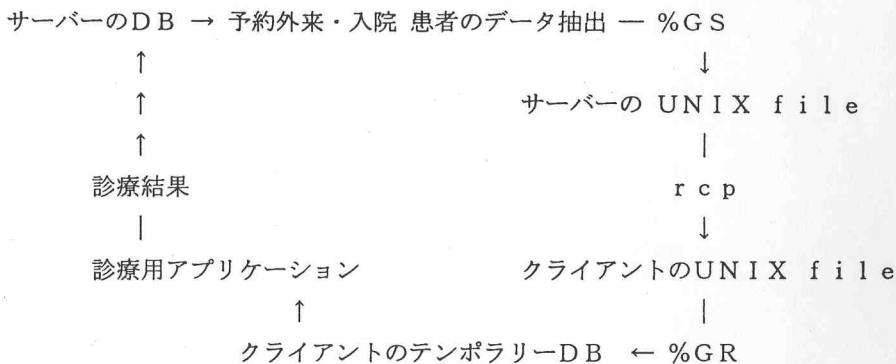
<実験> NETで363KB、%GEで592KB、件数2854件の容量のグローバルを
図1のようなルートで、CUP間を転送してみた。(回線の負荷が最小限のとき)

結果 : %GS=4秒 r c p=4秒 %GR=12秒 合計 = 20秒
 : 同じデータをDDPによりクライアント側からコピーした場合 = 73秒

外来診療に必要なデータサイズ(検査結果、投薬内容、予約情報、などを含む)は1患者当たりこれを上回る事はないと思われる。一日の患者数を30人と仮定しても最大10分間で転送できる。%GRにやや時間がかかるものの、r c pの早さが生きる。

この方式では、データベースの一貫性に細かく配慮しなければならないので、注意深い管理が必要とされる。ローカルデータベースの更新や削除のタイミング、即時性を要求されるデータと履歴データのように1日程度の時間差が許されるものとを丁寧にきり分けるなどが求められよう。

図1. テンポラリーデータベースの構築法



5. おわりに

ネットワーク環境でのM言語によるシステム構築について考察した。ダウンサイジングの波、オープン化の流れ、それに情報ハイウエーやマルチメディアなど技術環境はネットワークを前提としたものである。M言語がこの環境で生かされるためには、言語仕様だけではなく、このような外部環境への適応がいかに充実しているかが問われる。同時にまた、個々の応用システム構築のために、効率の高い技法を確率することも欠かせない要求の一つである。

筑波大学附属病院における病歴資料・情報の 共有化と情報システム

筑波大学臨床医学系 高田 彰

筑波大学附属病院の病院運営にはいくつかの特徴あるシステムが採用されている。その基本的な考え方は、病院の資源（人、予算、物品・設備・そして情報）の共有化を図ることにより病院運営を合理化し、患者を中心とした医療体制を提供することにある。情報の共有化という点では、病歴資料を病歴室で中央管理して院内公開制を保証しており、その管理対象も外来・入院診療録のみならず、レントゲン・フィルム、心電図、音図、脳波、超音波画像フィルム、内視鏡フィルムなど多岐にわたっている。入院中の患者についても、入院診療録を除き病棟には病歴資料をおかず、病歴室において管理を行い、各種のカンファレンス等も病歴室で複数の専門家をまじえて行っている。各種の検査資料については、必ず読影レポートが添付されており、診療の水準が常に高いレベルに保たれるようになっている。

病歴室における業務を担当している職員は10名であり、極めて多忙であることから、病院情報システム高度に利用して業務の合理化を進める必要がある。また、最近では各種の画像診断装置と連動して画像情報システムが導入されるようになり、病院情報システムとの接続が課題となる。これらの点を考慮し、本院中央・特殊診療部門での情報システムの構築と運用を行っている。

本院では一患者一ID一病歴（親袋）制で病歴資料の管理を行ってきたが、各病歴資料の管理をシステム化するにあたって、各病歴資料にも固有のIDを与えることとした（外来診療録を除く）。各病歴資料用IDは、入院診療録については病棟における端末で入院登録が行われた時点で、各検査資料に関しては部門において検査受付が行われた時点でバーコード化され発行される。放射線検査、機能検査、内視鏡検査では、各種診断機器と部門情報システムが接続されており、ここでバーコード化された情報を検査機器側で読み込むことにより、病院情報システム側の情報を検査機器側に転送することが可能となっており、円滑に検査を行うことに役立っている。放射線診断機器との接続では、機器側より使用したフィルムのサイズと枚数など情報が転送され、照射録の作成や、医事会計システムへの情報転送なども行われる。各種資料は病歴室にられるが、病歴室での受け入れと各種診断レポートの作成について情報を管理しており、その情報入力にしても資料用IDを用いており、業務の改善に役立っている。

各種の画像診断装置を含めた多様な情報機器が相互に接続されて情報システムが形作られるようになり、生ずる情報の種類と内容が多様化し、その蓄積も分散化している。このような状況において、分散化したシステム間で情報の整合性を保ち、物としての病歴資料と情報との対応をも保つ必要があると考えられるが、本院のシステムではこの点について、ある程度の成果をあげていると考えている。

参考資料

「筑波大学附属病院における診療資料の管理について」 診療録管理 Vol. 2, No. 2:40-43, 1990

「放射線部門情報システム(RIS)の検証 なぜ必要か、なぜ普及しなかったのか」

INNERVISION Vol. 8, No. 4:2-5, 1993

北米MTA報告

福井医科大学医学情報センター

山下 芳範

今年は6月13日から17日まで、ネバダ州リノ市で開催されました。会場はリノ市郊外のコンベンションセンターで行われました。

今回の大会では、かなり多くのチュートリアルが用意され、テクニカルな発表と並行して行われています。会場の2/3ほどが、このチュートリアルの割り当てられています。特に新しい仕様（主にGUIなどの新装備）に関する解説も含めたチュートリアルが人気があったようです。

今回の主な内容は、

チュートリアル

MWAPI (ウィンドウ)

V A fileman と kernel

ANSI M プログラミング

新装備入門

テクニカル及びQA

VAの新しいシステムについて

画像および音声アプリケーション

開発ツール

各社Mの発表

SQL関連アプリケーション

DHCP

電子化診療記録について

新技術の紹介

サーバー・クライアント

オブジェクト

です。特にQAができるように時間などが配慮されています。

今回は全日にわたって8つのチュートリアルが並行に行われており、この中でも、話題の新装備やGUI、SQL、ネットワーク等も含まれています。チュートリアルは\$200前後ですが、ほとんど全てが満員とのことでした。

展示に関しては、主なベンダーは例年通り参加していますが、どこもGUIを中心とした展示です。

MDC 報告 —MDC update—

福井医科大学医学情報センター

山下 芳範

今年は、2月24日から27日までテキサス州ヒューストン市で、6月9日から12日までネバダ州リノ市で開催されました。

今年は新しい規格案がANSIに送られましたので、94年規格としてANSIへそしてISOと成立するものと思われます。

この新しい規格はどのようなものかを、この規格案を中心に報告を行います。

この規格より新しく装備された主なものは、

s v n (システム変数)

\$DEVICE

\$ECODE、\$ESTACK、\$ETRAP

\$KEY

\$TLEVEL、\$TRESTART など

s s v n (構造化システム変数)

^\$CHARACTER

^\$DEVICE

^\$GLOBAL

^\$JOB

^\$LOCK

^\$ROUTINE

^\$SYSTEM

関数

\$REVERSE、\$STACK

\$QLength、\$QSUBSCRIPT

命令

MERGE、トランザクション処理

コントロールモニター など

その他

OMI (X11. 2)

GKS (X11. 3)

MWAPI (X11. 6)

MWAPIでは ^\$WINDOWが使われる。

です。これらの新装備に概要を解説します。

M Technology in Brazil: Status Report

マルシオ ビチキ ド アマラル Marcio Biczuk do Amaral

千葉大学医学部附属病院医療情報部

Division of Medical Informatics, Chiba University Hospital

The history of M in Brazil began in 1974, when *Biodata Informatica e Tecnologia* started the automation of a medical laboratory. M started gaining momentum in 1978, when COBRA (*Computadores Brasileiros*), after an agreement with Biodata and Meditech, create a MUMPS dialect, and became the first implementor of MUMPS outside the United States. Since that time, the use of M grew enormously in all sectors of the Brazilian market. Actually, the biggest user is the government. Governmental applications predominantly run under MUMPS/VM on IBM mainframes. Other big users are the three main oil companies (Petrobras, Shell and Esso). Petrobras also uses MUMPS/VM to control and process its fuel-distribution system. Another important user is the TV Globo, the nation's biggest television network. It uses M, running under VM for recovery of journalistic images and texts. The MTA Brazil has been working intensively in order to keep the strong tradition of M in Brazil, and at the same time to promote the image of M as an evolving technology. [1]

The annual meeting of the MTA Brazil has been made for 15 years. In the last years, this event was performed together with the Fenasoftware (*Feira Nacional de Software*), that is the most important computer industry event in Latin America, and considered among the world's three largest (ref. MTA North America, MUMPS NEWS, 1992;9;4:p5). About 1020 companies participated at Fenasoftware 1993. In the last year's event, the MTA attracted 220 new members and had more than two thousand visitors to its demonstration booth. The group also sponsored panel discussions and presentations to showcase M Technology. In 1992, the MTA-Brazil started publishing a quarterly issue of the M Technology NEWS. This publication contains up-to-date topics related to the M world, advertisements of companies and the characteristics of their products, the market, and events. On the table shown below there are some examples of Brazilian companies that are vendors or implementors of M. Some companies, like CompScientia, SEC and Medidata have cooperative agreements with American companies. Other companies, like Pensamento, Extensao and Ipsum, that produced national dialects of M in the past, share the market of Brazilian-made computers. They also furnish versions for PC/DOS and UNIX environments. Other main vendors of M in Brazil are Digital Equipment Corporation with DSM, and IBM, that has a big market share regarding versions for mainframes. [2]

<i>Brazilian Company</i>	<i>USA Partners</i>	<i>HARDWARE (Platforms: DOS,UNIX,RISC,VM,Other)</i>
CompScientia	Micronetics	IBM-Mainframes,M680xx,PC compatibles,Others
SEC	DataTree	PC compatibles,WS,VAX,Others
Medidata	Intersystems	Medidata M1001*,M386sx*,M486*,Labo,PCc,Others
Pensamento	DataGeneral	Cobra Cx*,Eclipse,Bull,Apple,DataGeneral,PCc,Other
Ipsum	Intersystems	Digirede8000*,M680xxArchitectures,PCc,Others
Extensao	Motorola	Cobra X10*,SCO*,Motorola,AT&T,DataGeneral,HP,PC

PCc=PC compatible; WS=WorkStations; M680xx=Motorola architecture; *Brazilian-made computers; Ot=Others

References:

- [1] Paulo Roberto Baratta (President of the MTA-Brazil). M Technology: Very Much a Part of Brazilian Commerce. In *M Computing* 1994;1(1):13-15.
- [2] MTA-Brazil. MUMPS NEWS, *Informativo da M Technology Brazil*, March93 and June93.

MUMPS Market Assessment and Opportunity

Bob Mappes

Micronetics Design Corporation

Two years ago MTA-NA sponsored and received a market study by the prestigious Gartner Group entitled *MUMPS MARKET ASSESSMENT AND OPPORTUNITY*. This report, funded primarily by the MTA-NA and the M implementors, indicated that the world-wide market for M and M related products was \$1 billion in 1991 and, providing certain actions were taken by the M community, would double in size to over \$2 billion by 1996.

In this projections for the growth of M, the report stated that "the highest rate of growth for M is being viewed outside the United States". The report went on to say "areas such as Eastern Europe and South America have become locations needing the basic capabilities and applications currently available on M". The section titled **Geographic expansion** contained the following: "Movement into Eastern Europe, South America, and the Pacific Rim will continue to add growth to the M market. With 45% of the revenue coming from non-United States sales in 1991, M has established a new base for growth. We would estimate that over 50% or growth will continue to come from outside the United States for the next 3 to 4 years".

The Gartner group did not have sufficient information on the M market in the Pacific Rim to consider the potential this extremely large area represents.

Mr. Mappes will discuss issues that the Gartner report identified the M community must address if M is to achieve a 2\$ billion annual revenue figure, where the M community stands today regarding these issues and, based of what has been learned in the US M environment, offer suggestions for the Asian M participants that could lead to a wide acceptance of M across all industries and lead to achieving an even larger annual revenue figure than 2\$ billion for M.

E s i O b j e c t s の概要

EsiObjects Overview

今泉幸雄
Y.Imaizumi

EsiObjectsアプリケーション開発環境はPCのウィンドーズ上で動いて、オブジェクト指向(OO)とM言語をベースにしてる。EsiObjectsとM言語の主な違いは、EsiObjectsはオブジェクト指向であるので、メッセージの文法、カプセル化、データ抽象化とオブジェクト・タイプ、ポリモーフィズム、継承等の機能がある。EsiObjects開発環境は、オブジェクトブラウザやユーティリティーズの形式のプログラム作成道具である。EsiObjectsにおける継承、ラベルや変数の有効範囲とメソッドの概念を述べる。継承の考えはプログラミングの道具としてはかなり強力であり、コードやデータの再利用性に優れてる。全てのクラスはライブラリにあり、標準で準備されてる。

(キーワード : EsiObjects、オブジェクト、クラス、インスタンス、メソッド、継承)

The EsiObjects Application Development environment is based on the OO(Object Oriented) and M language under PC windows technology. Of course, the major difference between EsiObjects and M language is that EsiObjects is OO, has a special message syntax, enforces encapsulation, data abstraction and object typing, polymorphism, inheritance and etc. The EsiObjects development environment provides a full set of programming tools in the form of object browsers and utilities. I am going to explain the concepts of methods, method labels and variable scoping, within the concept of inheritance in EsiObjects. Inheritance is the mechanism that gives the OO paradigm its power as unifying application development tool. It promotes reusability of code and data. All classes belong to a library and ready-made already.

(Keyword : EsiObjects, Object, class, instance, method, inheritance)

サンド薬品株式会社 筑波総合研究所
〒300-33 つくば市大久保8番地
(tel: 0298-65-2205)

SANDOZ Pharmaceutical Limited
Tsukuba Research Laboratories
Ohkubo 8, Tsukuba-shi, Ibaraki 300-33

1. はじめに

オブジェクト指向の考えが最初に論文に発表されてから20年になる。最近ではプログラミング言語や人工知能だけでなく、ソフトウェア全体に影響を与えて、標準化もすすめられて実用化された製品が多い。勿論、M言語もMDC(M Development Committee)を中心に標準化が進んでいる。EsiObjects(米国のEducational Systems, Incの製品)はDTM-PC(日本ダイナシステム(株)が販売とサポートするPC版M言語)上で動作する製品のひとつである。本稿は、EsiObjectsの動作環境、EsiObjectsのオブジェクトの概念と継承、標準M言語との違い、アプリケーションの利用と使用感想について述べる。

2. EsiObjectsの動作環境

2.1 ハードウェアとソフトウェアの条件

ハードウェアの条件としては、8MB以上のRAM,20MB以上の空ディスク領域,80486プロセッサ20Mz以上である。ソフトウェアの条件としては、MS-DOS Vesion5.0,Windows3.1のエンハンスモード,DT-MAX Vesion4.3M(With DT WIN1.2 サポート)以上,DT-Windows Version 1.2A 2以上,DTM-PC Vesion4.3M,EsiObjects Vesion1.0である。

2.2 EsiObjectsのスタートアップとシャットダウン

MS-DOS V5.0をスタートアップ後にDOSのプロンプトの状態、MWIN(MSWindows3.1とDTWindows)と入力すると、EsiObjectsV1.0のウインドーが開かれる。ここで、EsiObjects V1.0のアイコンをクリックすると、User Informationのウインドーが開かれて、UserIDとPasswordを入力する。Application Launch Pad(ALP:EsiObjectsのアプリケーション開発環境)のウインドーがあらわれ、File、Tools、Utilities、Options、Helpの5つのメニューバーがあらわれる。これよりEsiObjectsによるアプリケーション開発等が出来る。EsiObjectsのシャットダウンは、FileメニューのShutdownコマンドをクリックすると実行される。

2.3 EsiObjectsのアプリケーション開発環境(ALP)

ALPにはFile、Tools、Utilities、Options、Helpの5つのメニューバーがある。その中で主なHelp,File,Toolsについて述べる。

(1)Helpは教育用を含めて、OOP(Object Oriented Programming)に関するターミノロジー(オブジェクト、クラス、継承、カプセル化、インスタンス等)の解説とEsiObjectsの自習機能と操作中のヘルプ機能等から構成されてる。

(2)Fileは開発したアプリケーション・プログラムの実行、作成宣言、終了、保存、シャットダウン等から構成されてる。

(3)Toolsはクラス・ブラウザ、オブジェクト・ブラウザ、トリー、プロトコール、レイアウト、シェルから構成されてる。クラス・ブラウザ、オブジェクト・ブラウザは既にReady-madeされてるクラス、オブジェクト(一般のOOPのインスタンスと解釈するとよい)の参照機能、トリーとプロトコールはクラス等を階層構造にて表現する機能、レイアウトは再配置機能、シェルは直接にクラスやオブジェクトにメッセージ・パッシング(Message Passing:一般のOOPではメッセージ通信と呼ぶ)をDOコマンドで実行する便利な機能である。本稿ではこのToolsの機能を利用して、EsiObjectsのオブジェクトと継承を述べる。

3 オブジェクト

オブジェクトは変数(データ)とメソッド(動作の記述)がカプセル化(一緒になってる)されている。オブジェクトはクラス・オブジェクト(以後クラスと呼ぶ)とインスタンス・オブジェクト(以後インスタンスと呼ぶ)の2つに分けられます。ここではEsiObjectsにおけるOOPIについて述べる。

(図1 : Browser for the class Object)

3.1 オブジェクトの定義

EsiObjectの最上位クラスをスーパークラスと呼び,"Object"と言う名前になっている(著者はこの名前は大変誤解を招くと思われる)。クラスは定義の世界であり辞書の役割をする。インスタンスは現実の世界であり実際に存在する物を対象とする。例えば哺乳類や犬とはクラスであり、Aさんの家のポチはインスタンスである。EsiObjectsでは多くのクラスがReady-Made(標準で準備されてる)であり、利用者定義のクラスは全てスーパークラスであるObjectの下にリンクされる。インスタンスはクラスに実際の値が入って生成(定義)され、その時にOID(Object Identifier : オブジェクト識別)が自動的にかつユニークな識別番号が貸与される。

3.2 データの抽象化におけるクラスとインスタンス

例えば、Aさんのポチ、Bさんのラッシー、Cさんのケンタという犬の特徴をまとめると、"飼い主の犬"という特徴が抽出される。具体的には飼い主、名前があり、足が4本、人に懐く等である。ポチ、ラッシー、ケンタは、"飼い主の犬"というクラスのインスタンスである。ここで考えられるのはクラスとインスタンスの関係は1:nである。現実世界では複数のインスタンスからクラスを抽出する場合もあり(犬と飼い主の犬)、逆にクラスから複数のインスタンスを生成する場合もある(数学の定理と実際の方程式)。

3.3 オブジェクトの階層構造

オブジェクト(クラスとインスタンスの両方を含む)は階層構造になっている。Objectと呼ばれるスーパークラスが一番上に存在して、その下位のクラスをサブクラスと呼ぶ。サブクラスの関係には上下関係である親クラスと子クラス、同じ親クラスから生成された子クラスの同等関係をピアクラスと呼ぶ。インスタンスはあるサブクラスから実際の変数(データ)に値等が入って生成されて、最も下位のクラスに階層構造上リンクする。

3.4 クラス構造

クラスはクラス・メソッド、クラス・変数、クラス・パラメータの3つから構成されて、カプセル化(一緒になってる)されてる。クラス・変数はデータで標準M言語の記述と同じであり、クラス・メソッドの中で使用される。クラス・メソッドはクラスの動作を記述するプログラムの処理部分で、標準M言語の記述と同じであり、クラスやインスタンスの生成・操作・削除等を記述する。クラス・メソッド、クラス・変数の両方ともブラウザをすると、階層構造上で、I(Inherited : 上位クラスから継承される)、S(Shared : 当クラスで定義してサブクラスで有効となる)、P(Private : 当クラスのみで定義して他には有効でない)の区別がある。クラス・パラメータはメッセージ・パッシング時に利用する。

(図2: The SolutionApp class 図3: Class and Variable Browser 図4: The Tree Window for SuperClasses of SolutionApp 図5: The Method Browser for [Start.SolutionApp])

3.5 インスタンス構造

基本的にはクラスと同じ構成をとるが、インスタンス・メソッドとインスタンス・変数の2つから構成されて、カプセル化(一緒になってる)されてる。クラスと各々の違いは、インスタンス・メソッドはインスタンスの動作のみに対しての記述であり、インスタンス・変数はインスタンス・メソッドの中でのみ使用される。従ってメソッドも変数も使用目的と有効範囲が異なるので、クラスのととは別々のものである。クラス同様にインスタンス・メソッドとインスタンス・変数の両方とも、階層構造上のInherited、Shared、Privateの区別はある。

3.6 メッセージ・パッシングとポリモーフィズム

クラスからインスタンスの生成、インスタンス間の通信、インスタンスの消滅等の全ての動作はメッセージ・パッシング(メッセージ通信)にて実行される。EsiObjectsでは全てのオブジェクト(クラスとインスタンスの両方を含む)間のやりとりはメッセージ・パッシングにて実行される。記述形式は次のとおりである。

[Method.Object.Keywords](Parameter List)

Method : _(アンダースコア)で始まるクラスのメソッド名あるいは、インスタンスのメソッド名である。

Object : 通信する相手のクラスの名前(アンダースコアで始まる)あるいは、インスタンスの名前である。

Keywords : メッセージ・パッシングの配布方法である。
オプションであり使用されることは少ない。

Parameter List : 相手に送る変数の値でありオプションである。

基本はキー付きパラメータであり順序はない。

Parameter1=Value1(,Parameter2=Value2,.....)

Parameterの存在チェック(\$DATA)はメソッドが実施する。

事例としては

```
S PARTDB=[_New._PartsDatabase](Rootnode="^xPARTS")
```

```
DO [Start.SAPP]
```

```
DO [Browse._SolutionApp]
```

上記でおわかりのように、EsiObjectsは一般のOOP環境と同じくメッセージ・パッシングはポリモーフィズムが確立してる。つまり異なるオブジェクト(クラスとインスタンスの両方を含む)を同一の通信方式(プロトコール)で動作させることが出来る。この柔軟さはプログラミングの再利用性等に大変メリットとなり、カプセル化されたオブジェクトにたいする機密性の保持にも有効となっている。

4 継承

メソッド、メソッドのラベル、変数の有効範囲の継承(インヘリタンス: Inheritance)について述べる。

4.1 単一継承と多重継承

親クラスが持っているメソッド、メソッドのラベル、変数の内容を、子クラスが標準で参照可能であり、新たに子クラスで定義する必要のない場合(デフォルト)を継承と呼ぶ。例えば親クラスが自動車で、そのサブクラスにトラックと乗用車の場合に、自動車クラスで定義してるタイヤの数4つは、トラック・クラスと乗用車クラスはタイヤの数はデフォルトで4つになる。このように親クラスが一つのを単一継承(Single Inheritance)と呼ぶ。

子クラスから親クラスをみた時に、複数の親クラスがある場合を多重継承(Multiple Inheritance)と呼び、両方の親クラスのメソッド、メソッドのラベル、変数を継承する。

4.2 メソッドとメソッドのラベルの継承

新しいクラスが生成される時に、親クラスのメソッドとスーパークラス(EsiObjectsでは"Objects")のメソッドが継承される。インスタンスが生成される時に、生成するクラスのメソッドとスーパークラスのメソッドが継承する。

メソッドのラベルはEsiObjectsの形式になっていて、メソッドと同じく継承される。

```
* {Label}^{Vector}
```

* : 上位クラスから継承してる意味であるパブリック(Public) ラベルをあらわす。

*が無いラベルは 標準M言語と同じプライベート(Private) ラベルをあらわす。

Label : 標準M言語と同じである。

Vector : メソッドの探す階層構造の位置を示す。

事例としては

```
DO *init
```

```
DO *init^$SUPER ($SUPERはEsiObjectsの特殊変数)
```

(図6 : Example of Method and Label Inheritance 図7 : The Source Browser Window)

4.3 変数の継承

標準M言語と同じく、変数の定義はSET命令でする。

```
SET Name=Value
```

定義された名前と値はシンボルテーブルに格納されて、オブジェクト(クラスとインスタンスの両方を含む)が生成された時に付与されるOIDと組になる。

```
SET Name=[_New_ObjectClass](Name="ObjectA")
```

ObjectClassからObjectAという名前のオブジェクトを生成(Newメソッド)する。

これから変数の有効範囲について述べる。標準M言語の仕様になってる、ジョブ(Job)範囲有効なローカル変数(Local Variable)とファイル等で使用するグローバル変数(Global Variable)の2種類がある。また1990年度標準M言語仕様、NEWコマンドのローカル変数も含めて、EsiObjectsのオブジェクト(クラスとインスタンスの両方を含む)とメッセージ・パシ

グを含めると次のような変数有効範囲図が描ける。

変数Name はEsiObjectsの有効範囲から命名方法を拡張して。

SCOPE%NAME(S1,S2,...,Sn)

SCOPE : 変数の有効範囲であり、英字一文字である。

T : Temporary 単にメソッドの呼びだしに使用

P : Parameter メッセージ・パッシングで使用するParameter Listの変数

I : Instance インスタンス・変数

C : Class クラス・変数

U : Universal NEWコマンドが仕様になる前の有効範囲と同じ。

全オブジェクト(クラスとインスタンスの両方を含む)に有効。

N : Named Pool スーパークラス(Object)の下位のクラスに保存する。

(図8: EsiObjects Object Oriented Execution Context 図9: Scoping of Variables)

4.4 インスタンス変数のライフスパンと格納場所

インスタンス変数のライフスパン(Lifespan)は動的(Dynamic)と静的(Static)の2種類ある。クラスからインスタンスが生成される時に選択される。動的は参照時に作成されて後に変更可能な値であり、画面の高さ・広さ等の変数に利用される。静的は全てのインスタンス生成時に作成されて変更不可能な値であり、タイトル等の変数に利用される。

インスタンス変数を指定する事により、特別な場所に格納出来る。勿論グローバル・ファイルであり、NAME(S1,S2,...,Sn)の表現となる。例えば^VESPARTS("TEST"),PARTS("TEST")となる。

(図10: The Locate Button and Location Fields)

4.5 特別変数と特殊関数

EsiObjectsがOOPを導入するにあたって、20種類の特別変数(Special Variables)と13種類の特殊関数(Functions)を追加した。ここではOIDに関する6種類の特別変数とParameter Listに関する4種類の特殊関数を述べる。

特別変数(Special Variables)

\$SELF : オブジェクトのOID
 \$SUPER : 現在のオブジェクトのOIDから継承の探索をはじめる
 \$RETURN : メッセージ・パッシングのメソッドの終了コード
 \$ENVIRONMENT : EsiObjects環境のオブジェクトのOID
 \$CALLER : メッセージ・パッシングを送信した相手のオブジェクトのOID
 \$CLASS : 現在のクラス・インスタンス
 \$MESSAGE : 現在のメッセージ・パッシングのメソッド名前

特殊関数(Functions)

\$EXIST : オブジェクトの存在チェック(標準M言語の\$DATAと同じ)
 \$PARAMETERS : ParameterListの中から順序とりだし(標準M言語の\$ORDERと同じ)
 \$GETPARG : ParameterListの中から特定のパラメータの存在チェック

5 EsiObjectsと標準M言語の主な差異

すでに必要に応じてEsiObjectsの拡張部分を述べてるが、ここではプログラミングに関する部分について述べる。

(1) NEWコマンド

ローカル変数にだけ使用可能である。Temporary,パラメータ,インスタンス変数等では使用出来ない。NEW X,Y,ZはNEW L%X,L%Y,L%Zは同等でありLはローカル変数を意味する。

(2) KILLコマンド

引き数無しのKILLコマンドは禁止される。

(3)ポインター

間接的指定のポインター指定が標準M言語で多く用いられるが、EsiObjectsはメッセージ・パッシングによりOIDで実施できる。間接的指定はメソッドの名前、Parameter list、XECUTEコマンド等にて利用可能である。

(4)特別変数と特殊関数の\$TEXT

(5)NakedReference

EsiObjectsでは使用出来ません。

6. アプリケーションの利用

```
Object ---Database ---- DemoDatabase --- PartsDatabase
      |                                     |=====>PARTSDB
      + Application--- DemoApp ----- PartApp -----SolutionApp
                                                    |=====>SAPP
                                                    +----MyApp
```

----- クラス関係 =====> インスタンスを意味する。

事例として、部品倉庫管理のPartsAppクラスとSolutionAppクラスが提供されていて、利用者がPartsAppクラスからMyAppクラスを生成したとする。MyAppクラスは生成時に、親クラスからメソッド、変数、クラス・パラメータが継承されるが、MyAppクラスでのShared,Privateのは新たに定義する。

```
S PARTSDB=[_New._PartsDatabase](RootNode="^xPARTS")
```

PartsDatabaseクラスにメッセージ・パッシング(NEWメソッド)にて、PARTSDBというインスタンスを生成した。データベースとしてはRootNodeにて定義する。

```
S SAPP=[_New._SolutionApp](PartsDatabase=PARTSDB)
```

SolutionAppクラスにメッセージ・パッシング(NEWメソッド)にて、SAPPというインスタンスを生成した。使用するデータベースはPartsDatabaseのと定義する。

```
D [Start.SAPP]
```

SAPPインスタンスにメッセージ・パッシング(Start メソッド)にて起動する。

変
成
ル
種

7. 使用の感想

著者はEsiObjectsをIBM ThinkPadを利用した、目的としてはどんなオブジェクト構造をしているかを中心に調査した。従ってEsiObjectsの直接のアプリケーション開発は出来なかった。

- (1) 標準M言語経験者がオブジェクト指向について習得しやすい
オブジェクト指向についての自主学習機能がついてる。
マニュアルが標準M言語と比較する記述例(特にメソッドのプログラミング)が多い
"EsiObjects Getting Started Guide" (1993)。
- (2) ツールが完備されてる開発環境
オブジェクト(クラスとインスタンスの両方を含む)の利用者による追加・変更・削除がマウスのウィンドー操作で大変容易にできる。
変数やメソッドの継承がトリー図(階層構造)により大変理解しやすい。
- (3) メソッドのプログラミングがしやすい
Ready-madeのオブジェクトのメソッドのプログラムが公開されてる。
オブジェクト指向の特別変数と特殊関数が豊富である。
- (4) マニュアルの事例の詳細性に欠ける
部品倉庫管理(PartsApp)の事例において、業務の前提条件やデータ内容等に不足がありインスタンスを生成するとき条件等に理解しにくかった"EsiObjects Getting Started Guide" (1993)。

8. おわりに

従来のM言語の特徴を生かしながら、オブジェクト指向の開発環境を取り入れたEsiObj

ctsの概念と実際のクラスやインスタンスの生成、起動、消滅とプログラムの再利用性について、調査した結果について報告しました。今後はEsiObjectsがDTMだけでなく、他のM言語での動作とM言語間でのオブジェクトの流通性が高くなれば良いと思ってます。

[謝辞]

今回EsiObjectsを調査するにあたって、ハードウェアとソフトウェア環境の全てを貸与してくれた、日本ダイナシステム(株)の嶋社長と鈴木取締役に感謝します。

参考文献

- (1) "EsiObjects Getting Started Guide" (1993)
- (2) "EsiObjects User's Guide" (1993)
- (3) "EsiObjects Programmer's Reference Guide" (1993)
- (4) 今泉幸雄 "MUMPSによるオブジェクト指向型データベース管理システムへの私見", 第19回日本MUMPS学会大会, pp71-76, 1992
- (5) 今泉幸雄 "これならわかるオブジェクト指向(手続き言語時代の技術者を迷わしたオブジェクトとは何か?)", 第20回日本MUMPS学会大会, 1993

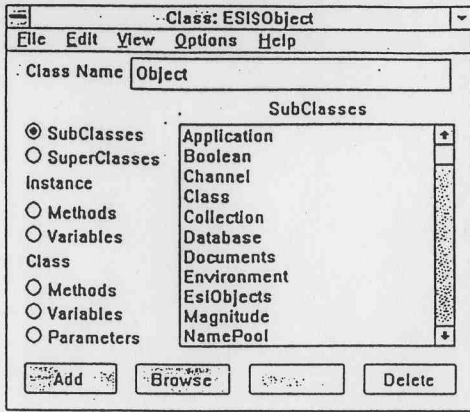


図 1 : Browser for the class Object

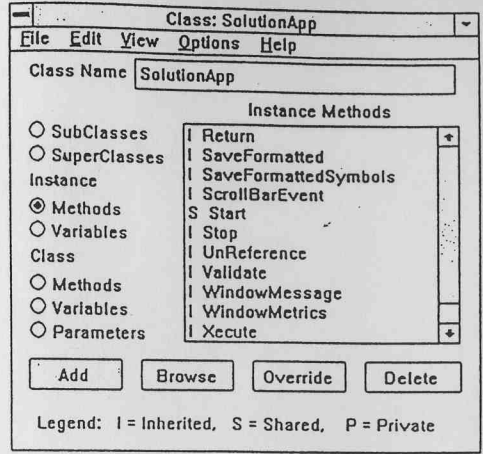


図 2: The SolutionApp class

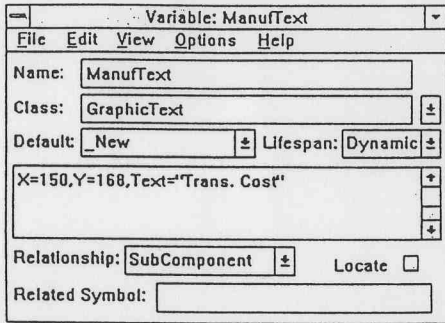


図 3 : Class and Variable Browser

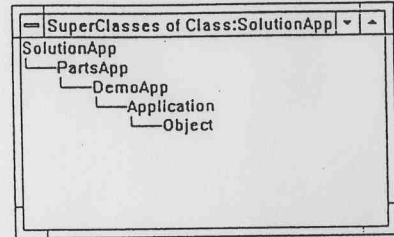


図 4: The Tree Window for SuperClasses of SolutionApp

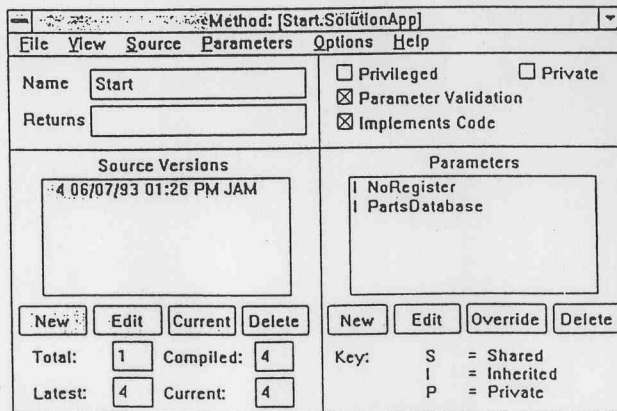


図 5: The Method Browser for [Start.SolutionApp]

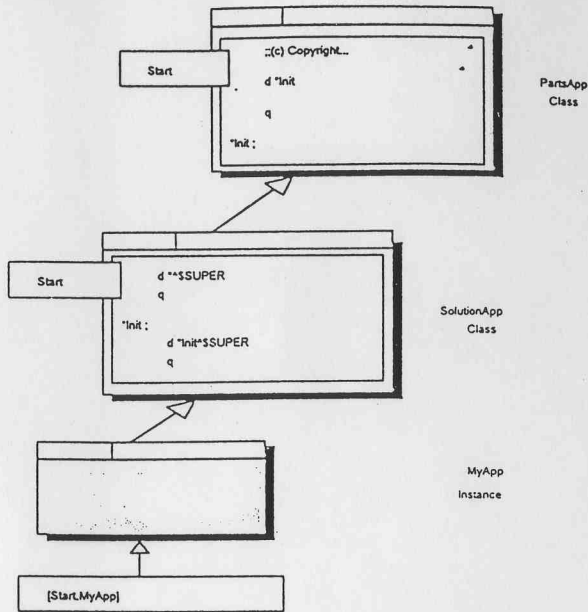


図 6 : Example of Method and Label Inheritance

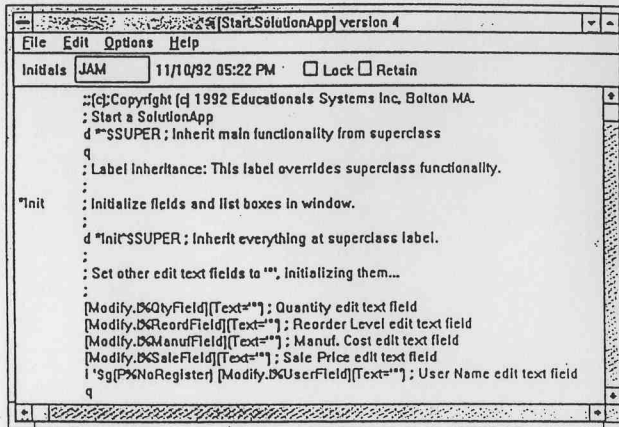


図 7: The Source Browser Window

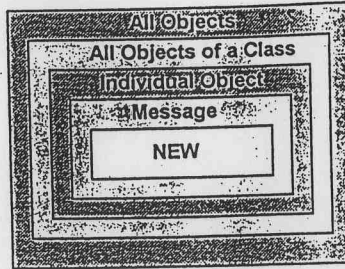


図 8: EsiObjects Object Oriented Execution Context

	Sub-routine	Message	Object	Objects of a Class	All Objects
New (1990 New'ed Local)	X				
Parameter	X	X			
Temporary	X	X			
Instance	X	X	X		
Class	X	X	X	X	
Universal (Traditional un-New'ed local or global)	X	X	X	X	X

図 9: Scoping of Variables

Variable: ManufField

File Edit View Help

Name:

Class: ±

Default: ± Lifespan: ±

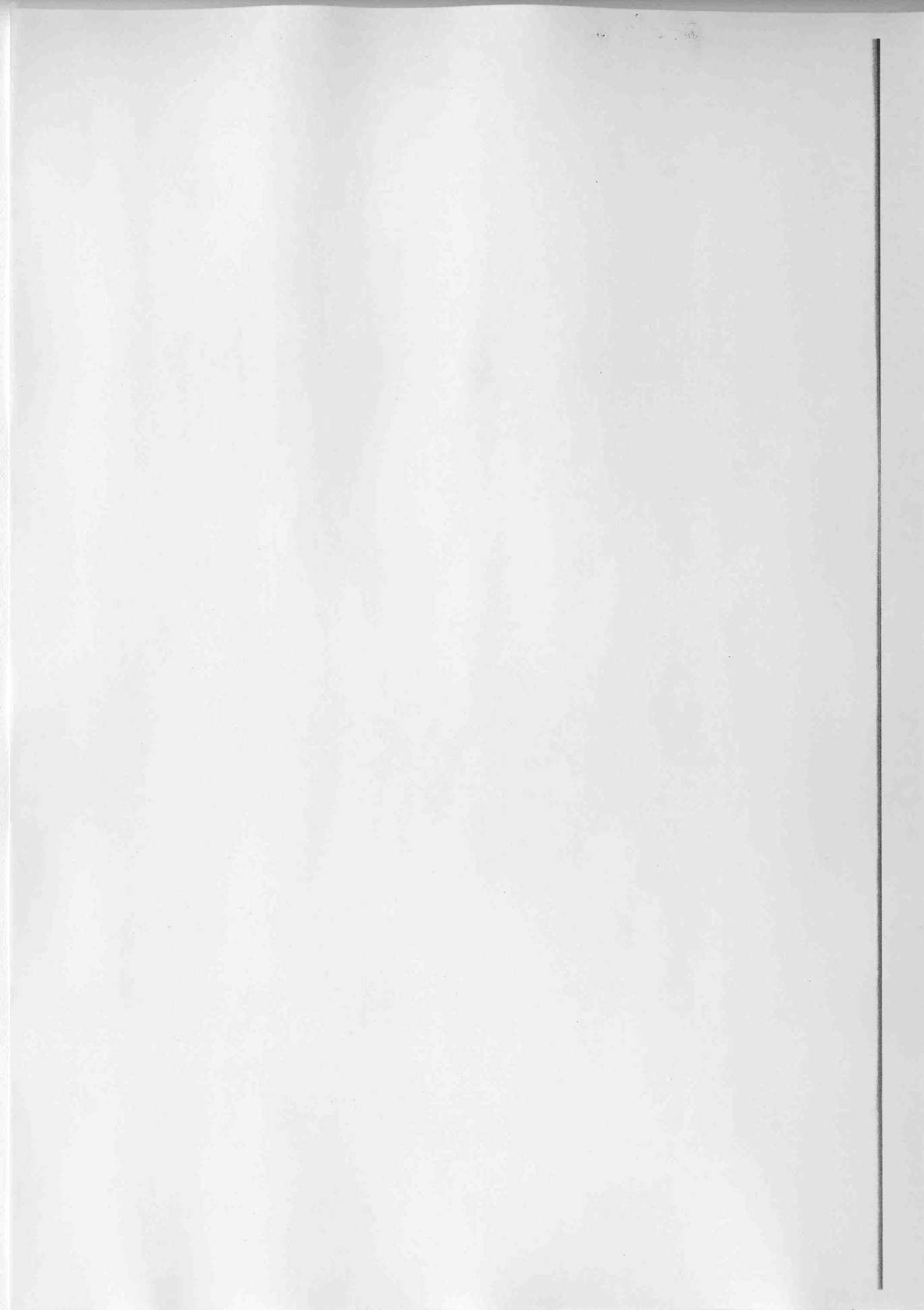
±

Relationship: ± Locate:

Related Symbol:

Location:

図 10: The Locate Button and Location Fields



一 般 演 題

千葉大学医学部附属病院に於ける、病床管理システムの導入について

○鈴木隆弘、里村洋一、本多正幸、山崎俊司

千葉大学医学部附属病院医療情報部

5/25
7/15 退院/卒
9/19 35.5日 稼働

千葉大学医学部附属病院では、病院情報システム「SYSTEM-CHIBA」を平成2年からスタートさせ、現在までに検体検査表示・検索、外来処方オーダー、外来診療予約、病歴管理、看護勤務スケジュール管理等のシステムが既に稼働している。平成6年5月より新たに、病床管理システムを導入したので報告する。

システム導入の目的

入院患者の情報を現場である病棟で入力することによって、迅速かつ正確な情報管理を行い、患者サービスの向上と転記等の事務作業の削減、病棟管理のために効果的な資料の提供などを行なう。更に、病棟システムとして今後導入予定の各種オーダシステム（処方、検査、放射線等）、病名システム等が稼働する基礎となる。

システム構成

Sunワークステーション上で住友電工のU-MUMPSを走らせ、データベースを構築した。端末には富士通のFMRシリーズを用い、同じく住友電工の端末エミュレータMWE80を走らせ、イーサネットを介してデータベースへアクセスする事とした。

システムの機能

主な機能を以下に示す。

1. 予約登録（入院予約、入院決定、転科）

入院が必要とされた時に入院予約を、入院日が決定した時に入院決定を登録する。

2. 病棟一覧

病棟にいる入院患者を一覧表示する。病床管理のメイン画面で、マーク表示や色分けにより、患者の現在の状態を表示する。

3. 入院登録／転入／転室登録、退院予定／確認登録

4. 外泊登録（外泊登録、外泊確認、帰院確認）

5. 看護情報入力

6. 病棟支援業務の各種リスト発行

システム導入の際の問題点

本システムは、他の病棟システムの基礎となるものなので、緊急入院などにも対応できるような迅速さや柔軟さのみならず、正確さも要求された。基本的には従来の手続きを踏襲して、手書きの伝票類を出来るだけコンピュータ上の帳票に置き換えて行く形でシステムの開発を行ったが、システム導入にあたって以下に挙げるような問題点が議論された。

1. 病棟関係

病棟と診療科は、多くの場合1対1の対応をしておらず、混合病棟や多病棟に跨がる診療科のために集計や一覧の操作を複雑なものにしている。

病棟一覧の画面は、種々の患者情報を表示して病棟内の患者の状態を一目で把握できるようにする事を目指したが、一画面に表示できる情報量の関係から断念した。

当院では、ICUに移動した患者は元の病棟にベッドを確保していなければならない取り決めになっているため、確保されているベッドをどう扱うかが問題となった。

2. 入退院関係

従来入院予約の手続きは各診療科によって大きく異なっていたため、全科に通用する予約手順と、予約の際に必要な入力項目を選択するのが難航した。

二重予約を防ぐ為に、入院が決定した患者をどのように表示するかが問題となった。

転科は病棟側の感覚としては入退院と同様の為、入退院と同じ予約手続きとした。

3. 看護関係

病棟管理日誌（日報）に出力する項目で様々な試行錯誤があった。

看護情報の入力にはコード入力による選択式を用いたが、そのコード表の内容を充分には煮詰め切れなかった。

4. 会計関係

病床/重症加算の情報は例外的な扱いが多く、病室番号からの自動入力は断念した。

会計システムは昼間のみ稼働しているため、曜日/時間帯によって病棟での操作方法が違う事となった。

考察

これらの問題点の原因は、病棟業務には緊急入院に象徴されるような例外的処理が多い事、病棟のスタッフにはコンピュータに不慣れな者が多かった事、十分な準備時間を持ってなかった事、コンピュータシステムの性能上の制約、複雑な組織によるものなどに分類できる。特に組織の面では、入院診療業務は医局、看護部、事務部という3系統の組織がそれぞれに縦割りかつ多対多で動いており、お互いの分担は入り組んでいる。このため従来の運用は大変複雑なものとなっており、今回の問題点の多くは、この従来の運用を大きくは変えないという前提で、システム導入を試みた事に起因すると考えられる。

今後のシステム導入では、状況に応じて旧来の運用をより合理的で患者本位なものに変えて行くことも必要と思われる。更に今回のシステム導入で、一部では運用見直しの気運が生れてきている事は、今後の新システム開発・導入にあたって良い傾向と言える。

福井医科大学新総合病院情報システムに おける M言語の利用と概要

福井医科大学医学情報センター

山下 芳範

福井医科大学では開院より約10年のM言語による病院情報システムを運用してきた。

昨年の概算で予算が認められたので、平成7年より新しい病院情報システムに切り替えを行う。

このシステムでは、ホスト系統を大きく増強するのではなく、端末系にもM言語を登載する新しい方式に切り替える。

概要

今回の計画では、オーダーエントリーの強化、電子化情報の推進、物流の組み込み、診療支援の強化を計画している。

また試験的であるがマルチメディアも取り込む予定である。

ホスト系は、2台を1組とした会計系・オーダー系・診療情報系の3系統およびバックアップなどを目的としたファイリング系統1台で構成する。

ディスク装置は、従来の3倍程度の容量であるが、低価格なSCSIディスクを用い信頼性向上のためにRAID5の構成としている。

また、トラフィックが比較的大きい診療系については、FDDI光LANで接続を行う。

端末系については、窓口等は、従来と同様に仮想端末としてネットワーク経由で接続を行うが、診療系についてはそれぞれの端末にM言語を登載する。

従来オーダー等の処理は、画面も含めてホスト系で処理を行っていたが、細かいサポート等で処理が増大しており、これ以上ホスト系を増強するだけの方策では、将来的にも高性能な機械に頼らざるえないので、端末側に分散することにした。

最終的なオーダー情報や診療情報は、従来通りホスト側に置き、画面系など負荷になる部分を端末系に配置する。

この方法により、ネット上のトラフィックは軽減され、ホスト系の負荷も軽減され、端末レスポンスも混雑時でも大きく変化しないものと予想している。

今回の計画では、端末が355台で、そのうちM言語を登載する端末は232台である。

これらは、ネットワークでDDP接続を行うこととなる。

端末側のOSにはWINDOWS-NTを中心に利用する。

端末のうち40台程度は、画像やマルチメディアの試験運用に利用する予定である。

今回の計画では、200台を超えるM言語登載機の相互接続を行うことになるが、日本ではまだこの規模の接続実績はない。

今回の計画を通して、今後のMの大規模分散環境での利用のためのデータとなるよう、今後は今回のシステムで得た情報を公開する予定である。

DEC 2100 (UMS) x 4 x 21 - (2)

DEC 3800 (11/17) P. 77°

端末 dpc 192 8

pcx66 123 8

DEC3300 40 8

島根医科大学教育研究システムの概要とシステム運用について

○笹川紀夫、山本和子、鎌江伊三夫
Norio Sasagawa, Kazuko Yamamoto, Isao Kamae

島根医科大学医療情報学講座
Department of Medical Informatics, Shimane Medical University

1. はじめに

近年、教育の場において情報教育の占める割合が大きくなってきており、医学教育の場においても例外ではない。

島根医科大学では1年生に「情報工学」、3年生に「医療情報学」として情報教育を行っており、特に1年生は後期に実習としてパソコンによるM言語教育を実施している。

従来の機器構成は、オリベッティのサーバー1台とサンヨーのパソコン5台をクライアントとし（いずれも16ビットAXパソコン）、NECのノートパソコン32台を端末として使用してきたが、処理能力や速度の点で十分な機能を発揮出来ず、また50名の学生に対して32台の端末と、数も十分ではなかった。

そこで本年4月、教育・研究用システムとしてワークステーション2台とDOS/Vパソコン50台を整備した。

今回は、教育・研究用システムの概要と運用について報告する。

2. 講義形態

本学では1年生の後期に「情報工学」の演習を約100名に対し実施しており、2班に分け、講義形態と演習の2部授業を行っている。

演習ではコンピュータのキー操作やM言語教育を実施している。

3. 教育・研究システムの概要

本システムの機器構成は以下のとおりである。

- | | |
|----------------|---------------------------------|
| A. ワークステーション | 2台 (DEC3000-800, DEC3000-600) |
| B. パーソナルコンピュータ | 50台 (DECpcLPv433dx) |
| C. プリンタ | |
| レーザープリンタ | 7台 (Canon LPB 6台, DEC LASER 1台) |
| ポストスクリプトプリンタ | 1台 (DECLN82R) |
| D. ネットワーク関連 | |
| HUB (8ポート) | 7台 (DECHUB90) |
| CSサーバー (8ポート) | 1台 (DECserver90M) |
| E. OS | |
| ・ワークステーション | |
| VMS | 1台 (DEC3000-800 7GB) |
| VMS、OSF/1 | 1台 (DEC3000-600 各1GB) |

M と クライアント/サーバーコンピューティング

M and Client/Server Computing

今井敏雄

Toshio Imai

日本デジタルイクイップメント(株)
西日本第一統合システム部

Digital Equipment Corporation Japan
Digital Consulting

〒530 大阪市中中之島2丁目2番2号
ニチメンビル

TEL(06)-222-9211 FAX(06)-222-9408

Recently, a new trend of so called End-User computing and Client/Server computing has emerged. This paper discusses how we should position M(MUMPS)— which was originally built as a user-oriented language by employing the then leading TSS technology— in light of this trend, and where it is headed in an emerging distributed application environment. I also mention that M Windowing API would be an innovative thing in that it not only raises the functional level of M from TSS CRT to GUI based, but signifies that M is beginning to change itself by absorbing advanced technologies to more positively interact with the outside world.

キーワード：M、クライアント/サーバー、DSM

はじめに

1967年、MGH(Massachusetts General Hospital)で誕生したM(MUMPS)は、TSS(Time Sharing System)という当時最先端の技術で、エンドユーザが自分達の問題を、自分達で"プログラム"し、解決出来る手段を提供した。70年中頃、日本のMUMPSの草分け京大病院では、事務官、技師、看護婦、医師自らがMUMPSを学習し、プログラムに取り組んでいた¹。解決すべき問題を抱えている人自らがコンピュータを使って解決するのがベストであるという考え

¹ 日本でのMUMPSの先達の一人である平川順名氏の当時のノートカバーに、"Do it yourself"というコピーが大書されていたのが印象的であった。

方がMの基本にある。そこから言語仕様、実行環境が設計されている。対話型(TSS)、インタプリタ方式、グローバル"変数"(ファイルでなく、ディスクにある"変数")、データタイプレス等々然り。もともと、"ユーザ・オリエンティド"な言語で、Ethernet LANが出る前から、DDP(Distributed Data Processing)という分散処理を当たり前のようになして来たMが、今日、"エンドユーザ・コンピューティング"(コンピュータの大衆化とエンドパワーの爆発)とそれをサポートしようとする"クライアント/サーバー"トレンドの中で、どう位置付けられ、どの方向に行くべきかを考察する。

トレンド

80年中頃から始まったPC(Personal Computer)の普及は、そのコスト性能比の飛躍的な向上と廉価高性能の市販パッケージの流布、GUI操作性の向上とあいまって、エンドユーザが、自ら問題解決に乗り出す大きな流れを生み出した。それまでの情報システム部門(IS)指導の集中型コンピューティング方式に対する、エンドユーザからの反乱である。そのPCトレンドと、"融通がきかない"、"時間がかかる"と言われようが業務をきちっと動かす使命をもつ基幹コンピュータシステムとのせめぎあい、融合が今行なわれている。クライアント/サーバー(C/S)コンピューティングは、分散したある意味で無秩序のPC群と、現存従来型システムとをどう結び付け、どう位置付け、どう進化発展させるかの試みである。また、企業が市場競争に勝つ為に、ビジネスニーズの変化にすばやく追従できる"柔軟性"のあるシステムが、そのC/S型コンピューティングに求められている。

Mの課題

トレンドの引き金は、CPU(チップ)コストの低下で、CPUをTime Sharingするまでもなく、CPUが安くなって来たことにある。PCは、1人1CPUの世界を可能にした。Mは、TSS技術の黎明期に生まれ、CRT(character cell terminal)をベースに発展して来た。Mの課題は、TSSからPC、C/S世代にどう飛躍革新出来るかである。エンドユーザ・コンピューティングとして先行していたMが、今、PCユーザに追いぬかれている。Mの魅力が、薄らいだとすれば、プログラムを(あまり)しなくても市販パッケ

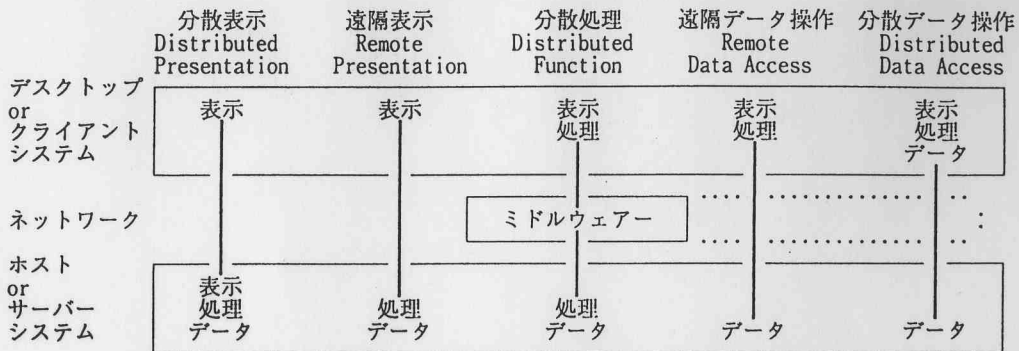
ージで、"Do it yourself"的に自らの問題を解決する手段がPC上に出て来たからではないか。問題を自ら解決したい人(Mはこういう人にアピールしていた)にとって、わざわざプログラミングせずとも問題が解決出来るなら、たとえ簡単にMがプログラム出来るとしても、Mは、ベストチョイスでなくなる。

しかし、PC上の"問題解決"は、あくまで個人レベルの解決である。グループや、会社レベル(Enterprise wide)の問題、分散データベースの問題、他システム/パッケージや基幹システムとの関係、統合上の問題は、エンドユーザに対してはまだ未解決である。"クライアント/サーバー"、"グループウェア"、GUI

簡易言語として復活した Visual BASIC、オブジェクト指向等々のテーマは、その未解決問題へのアプローチであり、いまだ万能薬が無い事の表れである。Mは、この今日のエンドユーザの問題にどう答えられ、答えていくかが問われている。Mは、あまりに自己完結環境にありすぎる。そのMの中だけに閉じ籠もることなく、外の世界とのかかわりの中で、Mの外のエンドの今抱えている問題にひとつの"答え"をMから提供出来なければならない。その為に、まず、MがC/S環境の下でどのような位置付けにあり、また、これからどのような方向に向かうのかを知る事が重要だと考える。

図 1

クライアント/サーバータイプと構成要素例



要素例

Motif/OSF
X-Window/X11
Terminal
Emulator
Printing
service
Form service
/FIMS(Form
Interface
management
System)
Graphic service
/GKS, PEX, PHIGS

ACMS Desktop

RPC(Remote Procedure Call)
OSF DCE RPC
Message Queuing
DECmessageQ
/OSI TPMQ
Object Broker
/OMG CORBA
(Common Object Request Broker)
Message service
DECMailworks
/ISO/CCITT
X.400
/XOPEN XAPI
EDI(Electronic Data Interchange)
/ANS EDIFACT
Management service
/EMA(Enterprise Management Architecture)

NFS(Network File Service)
File Sharing
DCE DFS
PATHWORKS
PC integration
Directory service
DNS(Domain name)
DCE CDS(Cell directory)
Distributed time service
DCE DTS

SQL Service
/ISO ANSI
DB integrator
ACCESSWORKS
Gateways
Data Distributor
Repository
CDD/Repository
/CDIF
Security Service
Transaction management
DECdtm
TP monitor
ACMS
ACMSxp/MIA
ODBC driver
(Open Database Connectivity)

DSM 要素例

M Window API
DECwindow/
Motif
interface

DSM API for
Windows
DDP DOS
DDE inter-
face
MS Win32
interface

DASL for ACMS
MIDDLE(M
Interface
Data Definition
Language
Extention)/M RPC.
Routine Translation
for Remote DO&JOB
DSM Namespace

DDP(Distributed Data Processing)
OMI(Open MUMPS Interconnect)

SQL/DSM
DB Integrator
Gateway for DSM

C/S と M の位置付け

クライアント/サーバーコンピューティングの目的は、メインフレーム、ミドル/ミニコン、ワークステーション、PC等の異機種(Heterogeneous)コンピュータをユーザ、アプリケーション間で透過的に統合/相互運用する分散アプリケーション下部構造(Infras-structure)を構築する事にある。クライアント/サーバーモデルは何を(表示部、処理部、データベース管理部)、どの程度(全部、一部)、どこに(クライアント、サーバー)分散するかにより、5つのタイプに分類される(Gartner Groupによる[1])。

各タイプと構成要素例(含弊社製品名)、および、DSM(Digital Standard M)のC/S構成要素例(含計画中)を図1に示す。

多(種)クライアントと多(種)サーバーの関係を柔軟で効率的に構成する為、クライアントとサーバーの2層(Two-tier)構成から、中間にもう1層、ミドルウェア(Middleware)と呼ばれる結合ソフト(Binding Software)を介する3層(Three-tier)構成のC/Sタイプへ移りつつある。このミドルウェアは、標準で高レベルのAPI(Application Programming Interface)を介して、クライアントおよびサーバーとインタフェースする。今後、透過的な分散機能を実現す上で、ミドルウェアは、重要な構成要素となる。

Mは、表示/処理/データ管理の各要素を、言語および実装上も一体化させる事によって、処理の最適化と簡素化、操作性と価格性能比の向上を図ってきた。これはMの使い易さの源泉であるが、逆に、80年代のコンピュータ機能要素の特化傾向の中で、Mが大きな革新を生めなかった技術的要因でもある。M外の世界では、データベース(RDB)のアプリケーションからの独立高機能化、X-WindowやMS Windows等の画面GUI特化高度化、計算サーバー、TPモニターなどの専門化が成果を上げ、今日、それらをどう有効に組み合わせるかという段階(C/S)にある。これに対し、MはTSS環境で既に完成度の高い自己充足したものを作りあげていたが、次への脱皮が難しい状況にあったのではないかと。

M Windowing API の意味

1993年MDC(MUMPS development Committee) Type Aから、いよいよANSI標準に上げる為のキャンパス(意見聴取/投票)に掛けられたM Windowing API(MWAPI)は、1970年代後半の、B(alanced)-Treeによるグローバル実装技術革新以来の快挙である。MWAPIは、M言語で上位レベルのGUI標準APIを規定する。言語標準で規定するという事は、プラットフォームやWindow OS(X-Window/Motif、MS Windows、Windows-NT、Macintosh GUI、OS/2 PM)が異なっても、ポータブルなMアプリケーションが書け、動くことを意味する。これは、他の言語にないMの大きな特徴となる。

MWAPIは、TSS CRTベースのMからGUI Mへの革新であるばかりでなく、Mが、自分でなんでもやる閉じた世界から、積極的にM以外の進んだ技術を取り入れ、脱皮発展しようとしていることを示している。SET、KILL、MARGE、\$OrderのMコマンドでアクセス可能な

Structured System Variable Names(SSVNs)

^\$WINDOW、^\$DISPLAY、^\$EVENT

の新設(Display属性、Window、Gadget属性、Event属性を添字構造にマッピングし意味付)、Call backルーチンによるイベント・ドリブン・ハンドリング、W / によるOPEN、USE、READ、WRITE型I/Oとの共存などに、今までのMの特徴、文化を壊さないうえに、いかに合理的に、外部のGUI成果をMの中へ取り入れようとしたか、その苦勞の程が伺える。MWAPIやSQL結合(Call in/out)での最近のMの動きは、Mの今後の方向性を示している。

M の方向

今後、Mの重要な方向として、図1の分散処理(Distributed function)C/S環境の中で、MサーバーあるいはMクライアントとして、存在機能が認められ、ひとつのFunction(アプリケーション)として機能して行く事だと考えます。ミドルウェアは、アプリケーションを統合するツールで、色々なアプリケーションが、それぞれ、その特徴を生かしながら、他のアプリケーションと協調して動く環境を構築することを目的としており、今後、こうした方向に進むと思われる。その中で、アプリケーションの1つとしてMが稼動する環境が、今後のMの方向ではないかと。

Mがこうした役割、位置付けを得る為には、Mとミドルウェアとの結合が必要となる。現在、ミドルウェアのアプリケーション統合方式として、以下の3つのタイプが有力である(図2参照)。

1. Object Request Broker

OMG(Object Management Group)でCORBA(Common Object Request Broker Architecture)として標準化が進められているもので、オブジェクト指向の概念をベースに、自律的なアプリケーション・オブジェクトに対するクライアント(またはサーバー)からの作業"Request(依頼)"の形態をとる。

2. Remote Procedure Call(RPC)

OSF(Open Software Foundation)のDCE(Distributed Computing Environment)で標準化が進められているもので、あたかも、Cのサブルーチンコールのように、別システムのプログラムを呼び出す仕掛けを提供する。

3. Message Queuing(DECmessageQ)

異機種プラットフォームのアプリケーション間で、上位レベルの同期・非同期タスク間通信を行うソフトウェア・メッセージ・バスを提供する。ネットワークング、エラーリカバリー等の詳細はミドルウェアが処理する。

表1に各処理方式の特徴をまとめる。

表1
分散アプリケーション・ミドルウェアの特徴

	ObjectBroker	RPC	MessageQ
通信目的	作業依頼	作業依頼	作業依頼 データ転送
構成	client-server peer-peer	client-server	peer-peer(対等通信) gateway
同期	一般に密	密	疎
信頼性	成功が失敗	成功が失敗	メッセージリカバー

上記方式の内、2.のRPCについては、Mからの繋ぎの試みは始まっている。1.、3.については今後の課題である。

こうしたミドルウェアが整備され、実現されて来ると、稼動するプラットフォーム、OS、アーキテクチャー(含データタイプ、コード系、言語)やネットワーク・プロトコルの詳細を知る事なく、上位レベルの標準のAPIで、他のアプリケーションと協調作業が可能となる。このAPIを通して、Mが外の世界と連繋していけるのではないかと。特に、Mを外と繋げる時の障害であるインタプリタ方式(実行時解釈)やデータタイプ変換の問題は、オブジェクト言語(C++等)での動的結合(late binding)の流れや、RPC IDL(Interface Definition Language)等のインタフェース・メタ言語の標準化で、その解決の糸口を得るかも知れない。ただし、Mは、宣言不要の言語で

あるので、インタフェース用のメタデータやグローバルの構造定義データ(スキーマ)をどのように(言語より上位のレベルで)持ち標準化していくかというMリポジトリの問題は、今後の重要なテーマとなると思われる。

おわりに

M誕生以来27年間、色々な言語が登場し、知らぬ間に消滅していく中で、Mは、創世時の"エンドユーザ"がプログラムし問題解決にあたる"実用的"な言語として、今なお健在である。DBMSの世界でも、リレーショナル型データベース以外でANSI標準の改定を今だ続けているのは、MのTree型データベースだけである。最近のM Windowing APIは、Mの革新であり、MTA(M Technology Association)の底力を感じる。

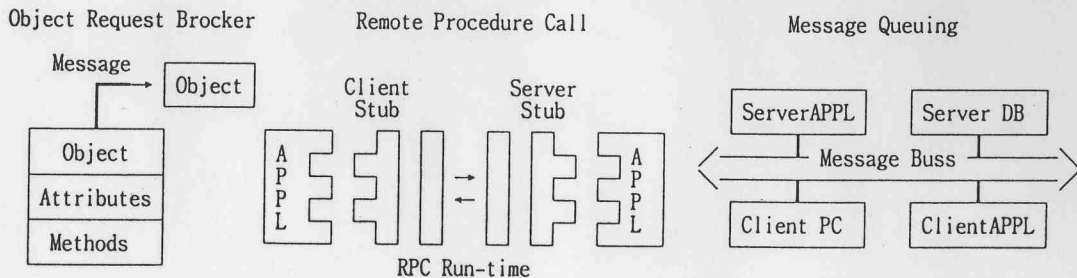
Mは、当初から、ユーザの視点で、"標準"にこだわり続けて来ている。Mの言語仕様の改定は、きわめてOPENである。言語改定は、継続参加条件はあるが、OPEN参加のユーザ、学識者、ベンダーがほぼ3分する、50から70人程のMUMPS開発委員会(MDC)とそのサブワーキング・グループで、十分に議論を尽くし、民主的な採決でTYPE C->B->A->ANSI/ISOへエスカレートする。それだけに、"なかなか決まらない"、"新しい事が自由に出来にくい"面もあった。しかし、やっと、動き出した感がある。外の世界から刺激を受け、Mの古い殻から脱皮しつつある。逆に、外の世界に刺激を与えるようになりたいものである。

文献

- [1] J.William Semich, "Where Do C/S Apps Go Wrong?" Datamation, January 21, 1994 : 30-36.

図2

分散アプリケーション・ミドルウェアのタイプ



Open環境で動作する新世代 DSM

A new DSM running on a open environment

前海好昭

Y.Zenkai

日本デジタルイクイップメント(株)
西日本第一統合システム部

Digital Equipment Corporation Japan
Enterprise Integration Center

〒530 大阪市中之島2丁目2番2号
ニチメンビル

TEL(06)-222-9211 FAX(06)-22-9408

DSM runs on open hardware platforms (such as, Alpha AXP, VAX, Intel) and open operating systems (such as OpenVMS, DEC OSF/1, MS Windows NT) and is ready for the new DSM generation aiming at establishing open access to M database and open client/server systems.

The following explains characteristics of DSM that runs on these new platforms.

要約

DSMは、オープンなハードウェアプラットフォーム (Alpha AXP, VAX, Intel, 他) オープンなオペレーティングシステム (OpenVMS, DEC OSF/1, MS Windows NT, 他) で動作し、Mデータベースのオープンなアクセスと、オープンなクライアント/サーバーシステムの構築をめざして、DSM新世代に入ろうとしています。

これら新しいプラットフォームで動作する DSM の特徴について説明します。

キーワード: DSM UNIX OSF/1 Windows NT Windows 4.0

DEC OSF/1 版 DSM の特徴

- Alpha AXP システム上で稼動 (完全 64Bit 対応)
- 1993 年度 TYPE A 言語拡張仕様の採用
- イ草案サポート
- ローカル変数、グローバル変数の最大文字列長 512 バイト
- サブスクリプト付きローカル変数、グローバル変数の最大長 245 バイト
- ルーチン行の最大長 512 バイト
- M ルーチン・ソース・コードの保護機能
- 外部呼び出し関数インターフェースの提供
- インクリメンタル・バックアップ/リストア機能の提供
- OpenVMS 版 DSM と機能的に互換性のあるシステム管理ユーティリティの提供
- DDP (分散データ処理) 機能の提供
- TCP/IP プロトコル標準サポート
- アフタ・イメージ・ジャーナリング機能の提供
- ビフォア・イメージ・ジャーナリング機能の提供
- X Window インタフェースの提供

Windows NT, Windows 4.0 版 DSM の特徴

- Alpha AXP システムと Intel PC 上で稼動 (Windows 4.0 は Intel PC のみ)
- 1993 年度 TYPE A 言語拡張仕様の採用
- イ草案サポート
- ローカル変数、グローバル変数の最大文字列長 512 バイト
- サブスクリプト付きローカル変数、グローバル変数の最大長 245 バイト
- ルーチン行の最大長 512 バイト
- M ルーチン・ソース・コードの保護機能
- 外部呼び出し関数インターフェースの提供

- インクリメンタル・バックアップ/リストア機能の提供
- OpenVMS 版 DSM と機能的に互換性のあるシステム管理ユーティリティの提供
- DDP (分散データ処理) 機能の提供
- TCP/IP プロトコル標準サポート
- アフタ・イメージ・ジャーナリング機能の提供
- ビフォア・イメージ・ジャーナリング機能の提供
- MS-Windows インタフェースの提供
- Microsoft Win32 interface の全装備
- MWAPI (M Windowing Application Programming Interface) の提供

おわりに

DSM は、Open な環境で動作する DSM に進化しつつあります。

弊社 OSF/1 版 DSM の登場は、DSM のマルチベンダー環境対応への第一歩となるものです。

今後、UNIX の世界は、SPEC 1170 で API(Application Programming Interface) が統一される方向に進むと思います。統一が進めば、あらゆる UNIX システム上で DSM を稼働させることが可能になります。また、Windows NT, Windows 4.0 のサポートは、世界に存在する 5 千万台以上のパーソナル・コンピュータ上で DSM が動作することを意味します。DSM を使用するためにハードウェアおよび OS を選ぶ必要がなくなる時代がもうすぐそこまで来ています。

異巾文字列処理の考察

String Functions for one-or-two Proportional Character String

馬場謙介

Kensuke Baba

国立埼玉病院臨床研究部

Department of Clinical Research, Saitama National Hospital

はじめに

LetterとTypeとから成る文字列に対し、単独で実際に文字列を取得できる文字列処理関数是用意されていない。(SZPOSITION関数は'知る'関数で、'得る'関数ではない)。従って、実際に分解結果を文字列で得ようとする、'知る'関数であるSZPOSITION関数と'得る'関数であるSEXTRACT関数との両方を組み合わせて使わざるを得ない。しかも、少し複雑になると、SZPOSITION関数とSEXTRACT関数の単純な組み合わせでは対処できなくなる。一般に、Proportional Characterの文字列処理と文字フォントとは不可分の関係にある。従って、使用する文字フォンの字巾辞書を定義しない限り、Proportional Characterの文字列処理は語れない。そこで、今回は、Proportional Characterの内、最も簡単なLETTER字巾:TYPE字巾=2:1の場合に限って考察する。この考察は、LetterとTypeとから成る文字列の文字列処理に適合した単独の(得る)文字列処理関数の仕様に示唆を与える。

範囲を指定した抽出

文字列Sを表示した場合の物理位置aから物理位置bの文字列部分を抽出する一般式を、

$$\text{SEXTRACT}(S, \text{SZPOSITION}(S, 1, a) + .5, \text{SZPOSITION}(S, 1, b) + .5)$$

としておくと、aやbが、Letterを中途半端に指定されても(不合理指定があっても)、不都合は一応回避できる。しかし、厳密には、aが不合理指定になった場合、結果の文字列は指定より半角分左にシフトしてしまっている。また、結果の物理長は、指定の物理長の±1半角巾の間を変動する不都合もある。

表示巾を固定した抽出

文字列Sを表示した場合の物理位置aから物理巾wだけ抽出する一般式は、

$$\text{SE}(\text{SE}(S, \text{SZP}(S, a) + .5, 1023)), 1, \text{SZP}(\text{SE}(S, \text{SZP}(S, a) + .5, 1023)), w) + .5)$$

で得られるが、この場合でも、結果の物理長は指定の物理長より1半角巾長くなることがある(長くしたくないのなら下線部分を削除)。ここでもシフトの問題が起こる。普通は、

表示を固定しシフトの問題とならないし、右端の超過もあまり問題にならない。

左端を調整した表示文字列の抽出

表示画面の複数行にわたり横スクロールしようとする等、表示文字列が左シフトしていたのでは上下に不揃いになって不都合である。不揃いを避けるために左端に表示巾がTYPE文字の中と同じFを充填するには、

```
$$($ZP(S,X)#1=0:"" ,1:F)_SE(S,$ZP(S,X)+1.5,1023)
```

とすると、目的の表示文字列を得ることができる。但し、X の定義は、(前項のaとは異なり)、X より左をカットする物理位置とする。

両端を調整した表示文字列の抽出

上を発展させて、両端を調整するには、仮想関数を用いる方が便利である。

```
$$BOTH^%CUT(String,X,x,F,f)
```

そして、この仮想関数は、

```
BOTH(S,X,x,F,f) d q F_S_f
.s:$zp(S,X)#1=0 F="" s:$zp(S,x)#1=0 f=""
.s S=$E(S,$ZP(S,X)+1.5,$ZP(S,x))
```

である。但し、X の定義の仕方は、(xとは異なり)、物理位置 x は、表示範囲内である。

X-Yフィールドへの表示

本論である「文字列Sを一定の長さの行に分割して、X-Yフィールドに表示すること」を考える。前の議論では、「文字の過長の問題」、「シフトの問題」は厳密に議論しなかったが、X-Yフィールドに表示するとすると、この問題を厳密に扱っておかなければならない。何故なら、前行の処理結果が次行に伝播するからである。そこで、この'伝播'をうまく扱うには、(\$PIECEが補助関数として\$FIND関数のように)、補助関数を定義して、これを使うのが常道の一つである。そして、n で指定した行が終わったつぎの(論理)文字位置を返す次の関数を用意する。

```
$$PACK^%FIND(String,w,n)
```

ここで、w はフィールドの物理横長)。この関数はワープロがそうであるように各行の始まりを左詰めにすることを想定した仮想関数である。この仮想関数の中身は、

```
PACK(STRING,w,n) n j,f s f=1 f j=1:1:n d q f
.s f=f+$l($e(STRING,1,$zp(STRING,w))),STRING=$e(STRING,f,1023)
```

である。この仮想関数(外部関数)を使うと、求める n 番目の行の文字列は

```
SE(String,$$PACK^%FIND(String,w,n-1,$$PACK^%FIND(String,w,n)-1)
```

となる。更に、上の値を返す仮想関数を

\$\$PACK^%PIECE(String,w,n,Filler)

を用意しておくこと更に便利であろう。ここで、Fillerは、右端が中途半端に切れたときの'詰めもの'で、通常スペースを指定する。

(ワープロ風ではなく)、両端を調整した場合のX-Yフィルドへの表示文字列についても同様に補助関数を用意して定義するのが賢明である。

言語要素を念頭に置いた単純化

外部関数として求められる仕様と言語要素としての関数に求められる仕様とは、同じではない。前者には、'直接の使用の便利さ'が後者に比べてより強く要求され、後者には、'言語として整合性'が前者に比べてより強く要求される。ここでは、言語要素に組み込むべき機能としての意味を強調して、よりスリムにした仕様を持つ外部関数を以下に示す。

```
%XX ;eXtended eXtended string functions for letter-type-mixed string (c)kb 1993.06.06
E(% ,a,b,j) ;EXTRACT
q:'$d(%)' "" q:'$d(a)' "" ;illegal
g:'$d(b)' LtBorder ;Left Border
s:'$d(j)' j=0 ;default
g:-j=2 LtBehind ;Left Behind
g:-j=1 LtBorder ;Left Border
q:+j=0 $e(%,$zp(% ,a)+1.9,$zp(% ,b)) ;Net String
q:+j=1 $s($zp(% ,b)#1=0:"" ,1:$e(%,$zp(% ,b)+.9)) ;Right Border
q:+j=2 $e(%,$zp(% ,b)+1.9,$l(%)) ;Right Behind
q "" ;----- ;illegal
EE(% ,a,b,j) n t ;EFFICIENT EXTRACT
q:'$d(%)' "" q:'$d(a)' "" ;illegal
g:'$d(b)' LtBord ;Left Border
s:'$d(j)' j=0 ;default
g:-j=2 LtBehind ;Left Behind
g:-j=1 LtBorder ;Left Border
s t=$e(%,$zp(% ,a)+1.9,$l(%)) ;work
q:+j=0 $e(t,1,$zp(t,b-a)) ;Net String
q:+j=1 $s($zp(t,b-a)#1=0:"" ,1:$e(t,$zp(t,b-a)+.9)) ;Right Border
q:+j=2 $e(t,$zp(t,b-a)+1.9,$l(t)) ;Right Behind
q "" ;----- ;illegal
LtBehind q $e(% ,1,$zp(% ,a)) ;Left Bihind
LtBorder q $s($zp(% ,a)#1=0:"" ,1:$e(%,$zp(% ,a)+.9)) ;Left Border
;===== ;
F(% ,w,n) n j,f,t ;FIND EFFICIENT NEXT FIRST POSITION
q:'$d(%)' 0 q:'$d(w)' 1 ;illegal
s:'$d(n)' n=1 ;default
s n+=n,w+=w s f=1,t=% ;inital set
f j=1:1:n s t=$e(% ,f,$l(%)) s f=$zp(t,w)¥1+f ;Main Loop
q f ;
;===== ;
P(% ,w,n) ;PIECE OF nth w-WIDTH EFFICIENT FIELD
q:'$d(%)' "" q:'$d(w)' "" ;illegal
s:'$d(n)' n=1 ;default
q $e(%,$SF(% ,w,n-1),$SF(% ,w,n)-1) ;Main
;===== ;
```

ここで、各行ラベルの機能は、

- E jの指定に従い、物理位置の文字または物理範囲の文字列を返す
- EE jの指定に従い、物理位置の文字または物理範囲の文字列を返す
- F 指定した序数の物理範囲の次の論理文字列位置を返す
- P 指定した序数の物理範囲に入る文字列を返す

である。EとEEの相違点は、前者が機械的に切取抽出するのに対し、後者は端に隙間が空いたときその隙間を利用することにある。また、受渡し変数の意味は次の通りである。

- % (処理の対象とする)主文字列
- a 始点の物理位置
- b 終点の物理位置
- w 抽出領域の物理巾
- n 領域の序数
- j 返値の種類(下参照)

jの値によって、返値は次のようになる。

- j=-2 左切断位置より左の文字列(不合理切断文字を含まない)
- j=-1 左切断の不合理文字(不合理でないときはヌル)
- j=0 指定領域の文字列(不合理切断文字を含まない)
- j=1 右切断の不合理文字(不合理でないときはヌル)
- j=2 右切断位置より右の文字列(不合理切断文字を含まない)

\$ZPOSITION関数の第3引数を活かして使うためには、第3引数の値を%ZPに定義して、上記ルーチンの\$ZP関数の第3引数として%ZPを追加することによって、上記ルーチンは、Letter巾:Type巾=2でない場合に簡単に拡張できる。

処理の種類の増加に対処するために、jによって処理の種類を指定するようにした。この方針は、「言語要素の数は少なく、夫々の機能は多様に」のM言語の設計の根本精神に叶っている。そして、jの値が、-2,-1,0,1,2 の意味の連想性も決して低くない。

考 察

M言語は、文字列処理に優れているが、Letter と Type が混在する文字列に対しては、'知る'関数として\$ZPOSITION と \$ZWIDTH が用意されているだけで、'得る'関数は、用意されていない。そのため、上記のような論じてきたように'知る関数'と'得る'関数である \$EXTRACT関数を併用しなければならない。Letter と Type が混在する文字列の処理は、日本特有の(正確には漢字を使用する民族の)問題である。議論を深めて、便利な仕様の'得る'関数を確立したい。ここで'言語要素としての関数'の仕様を意識して示した機能仕様が、次の次期の文字列関数の仕様の検討の土台になることを望む。

任意の文字巾を文字から成る文字列の Proportional 表示/印刷は、言語レベルで処理すべきか、アプリケーション・レベルの処理に委ねるかは、議論を要するところである。ここでは、この議論を避けて、LetterとTypeと成る文字列を対象に考察することにした。

GUIツール、MGRの概要と紹介

Introduction of MGR

○田久 浩志、小畑 恭弘*、中村 勝美*、志賀 晃*
○東邦大学医学部病院管理学研究室
*株式会社 九州ビット

1. はじめに

MGR/DOSV, MGR98は九州ビットの作成したGUIである。MUMPSの操作性を改良するために、MGRをメモリー上に常駐させ、ファンクションをコールすることにより、簡単な操作でボタン、ウインドウ、画像などを扱い、従来より指摘されていたMUMPSの操作性を改良するものである。MGRによりMacintoshのハイパーカードのような操作がMUMPSの中より可能になる。また、専用ボードと組み合わせ、JPEGの画像データを扱うことができる。本大会ではこのMGRの主な機能について紹介する。

2. 主な機能

ファンクションキー表示	マウスの制御	編集入力
グラフィック制御	アイコン制御	ウインドウ制御

使用するメモリーは、DOSの側に46Kbyte、EMS側に最低1Mbyteのメモリを必要とする。入力画面の作成にあたっては、画面の背景はグラフィックで作成し、指定した矩形の領域に文字列を入力する方式を採用している。またWindowsのビットマップをボタンに張り付けることも可能である。以下に、MGR単体を組み込んだルーチンを用いて、MGRの使用例を紹介する。

2.1 画面の作成

画面に表示する入力する項目には各々ID番号を設定する。このIDを元にして、入力領域であるインプットフォーム、参照領域であるリファレンスフォーム、入力内容の説明をするガイドフォームを設定する。

生年月日	<input type="text" value="入力領域"/>	<input type="text" value="年齢"/>	歳
生年月日、歳	ガイドフォーム		
入力領域	インプットフォーム (編集入力を行う)		
年齢	リファレンスフォーム		

2.2 選択入力

ウインドウを開き、CRで「はい」を、ESCで「いいえ」を選択する。選択されたアイコンは押したように表現される。動作確認の時に使用すると便利である。ウインドウの開く位置は上から5行目、左から4文字目で固定されている。MESに文字列を;で区切って与えると、縦に複数行で文字列、MESが表現される。

```

EVZANS; Example for VZANS
MAIN D INIT^VZSYS
CALL ;
      D INIT^VZMNU D SETUP^VZMNU
      s MES="CR=yes ESC=no"
      s junk=$$ANS^VZANS(MES)
EXIT D EXIT^VZSYS
Q

```

2.3 グローバルの選択入力

あらかじめ決められた形式のグローバルを対象に、それを表示し、選択することを行う。選択した結果はノードの値として帰される。対象とするグローバルのルートには"ウィンドウへの表示名称;コード幅;データ幅;表示最大数"の形式でデータが設定されているのが望ましい。そうでないときは、ウィンドウへの表示名称=データ選択、コード幅=10、データ幅=20、表示最大数=99999が設定される。

```

EVZSEL; Example for VSEL
MAIN D INIT^VZSYS
CALL ;
      D INIT^VZMNU
      D SETUP^VZMNU
      s ANS=$$SEL^VZSEL("MBSYS(100,11)") ;この前でデータを設定
EXIT D EXIT^VZSYS
Q

```

3. まとめ

MGRは九州ビットが独自に開発したGUIであるため、MWAPI等のウィンドウに対するハンドリングとは別規格のものである。しかしながら、閉じたシステムで使用するに特に問題は無いといえよう。MGRは機能が非常に多いため、少々使いこなすのが難しい印象をもった。単体のMGRに被せたツール類を開発することにより、それらの問題は軽減されよう。今後、ユーザーにとってMGRが使いやすい環境が、整えられることを希望するものである。なお、本MGRのPC98版の単体は、一般に公開されている。

連絡先：東京都大田区大森西5-21-16 東邦大学医学部 病院管理学研究室
Nifty-ID MAF00072

文書データの検索に関する考察

On Document Data Retrieval

馬場謙介

Kensuke Baba

国立埼玉病院臨床研究部

Department of Clinical Research, Saitama National Hospital

はじめに

箇条書きでも文章データとして使用する単語が自然語である場合の検索の手法に常道はない。多くの場合、データを端から逐次読んで該当症例を抽出する方法がとられている。しかし、これでは、グローバルのリファレンス機能を生かしたことはない。日本病理剖検輯報の日本語データベースでは、グローバルを生かすことを含めたいろいろの工夫によって検索の効率化をはかろうとしている。実際には、これらの工夫を組み合わせて結果として検索速度の改善がはかっている。検索速度の改善に役だった工夫につきいて論及する。

情報の統一

検索を単純にする目的で、誤字の訂正(表1a)、文字の統一(同義異字の統一、半角・全角の統一、ひらがな・カタカナの統一、種々括弧の統一、句読点の統一)、用語の統一(表1b)等を実施した。表の例で示した変換辞書はインテラクティブに生成したものである。現在その内容の充実をはかっている。また、単位表現を統一し、数量データは表2の論理に従って、参照データからほぼ自動的に削除した。

修飾語の扱い

修飾語等は、その性格(程度、分布、側性、並列、手術、置換術、人工臓器、放射療法、化学療法、免疫療法、移植、合併治療等)により処理を変えた(表3)。この過程で、例えば「胃癌術後」の記載から「術後」を削除した。しかし、逆に、「放射線術後大腸炎症」はそのまゝ残した。その判断はインテラクティブに(安全側に偏して)実施した。

Table 1a Miss-spelling (Exp.)

多発性脳梗→多発性脳梗塞	
多発性脳塞→多発性脳梗塞	
嵌屯→嵌頓→嵌頓	
総胆管→総胆管	末梢→末梢
心衰炎→心囊炎	硬塞→梗塞
為膜性→偽膜性	窒息→窒息
栗粒性→粟粒性	表弱→衰弱
機質化→器質化	内蔵→内臓
硬塞→梗塞	腫張→腫脹
懐疽→壞疽	透折→透折
未熟→未熟	表弱→衰弱

Table 1b Normalization (Exp.)

そううつ病→そう鬱病→そう鬱病	躁鬱病
びまん性→び慢性→彌漫性	彌漫性
タンパク→タン白→たん白	蛋白
るいそう→るい瘦→羸瘦	瘰癧
クモマク→くもまく→蜘蛛	膜
ギ膜性→ぎ膜性→偽膜性	膜性
胆のう結石→胆嚢結石	胆嚢結石
うつ血→うつ血→鬱血	鬱血
胆汁うっ滞→胆汁鬱滞	胆汁鬱滞
骨粗そう症→骨粗鬆症	骨粗鬆症
びらん→び爛→糜爛	糜爛
	頸→頸
	鬱→鬱
	總→總
	眞→眞
	ゆ着→癒着
	のう腫→囊腫
	乳ビ→乳び→乳糜
	胆のう炎→胆嚢炎

Table 2 Essence of Logic to Eliminate Quantitative Adjectives (Example)

摘 要	消去論理の要点 (MUMPS言語により記述)
経 過(1)	1N.N."全経過"."半"."年"."半".N."ケ"."ケ"."月"."半".N."日"."の"."状態"
経 過(2)	1N.N."半"."年"."半".N."ケ"."ケ"."月"."半".N."日"."経過"."の"."状態"
計 量 値(1)	."重量"."".1N.N1"g"."r"
計 量 値(2)	."右"."左"."両"."側"."約".1N.N1"L"."."
計 量 値(3)	."右"."左"."両"."側"."約".1N.N1"l"."."
計 量 値(4)	."右"."左"."両"."側"."約".1N.N1"mL"."."
計 量 値(5)	."右"."左"."両"."側"."約".1N.N1"mL"."."
計 量 値(6)	."右"."左"."両"."側"."約".1N.N1"cc"."."
計 量 値(7)	."右"."左"."両"."側"."約".1N.N1"輻"."."
計 量 値(8)	."大きさ"."直"."半"."周"."径"."長"."約".1N.N1"cm"
数 量	(数個、数ヶ、数ヶ、数コ、多数、小数、少数)
死後時間(1)	"死後"."約"1N.N"時間"."の"."状態"
死後時間(2)	"死亡"."約"1N.N"日"."約"."約"1N.N"時間"."約"1N.N"分"."後"."の"."状態"
胎 齢(1)	"生後"."約"1N.N."日"."約"1N.N."時間"."約"1N.N."分"."の"."状態"
胎 齢(2)	"胎齢"."約"1N.N."ケ"."ケ"."月"."約"1N.N."週"."約 "1N.N."日"
胎 齢(3)	"胎令"."1N.N."ケ"."ケ"."月"."約"1N.N."週"."約 "1N.N."日"
胎 齢(4)	"在胎"."約"1N.N."ケ"."ケ"."月"."約"1N.N."週"."約 "1N.N."日"

Table 3 Eliminated Adjectives (Exp.)

種 別	処 理	例
程 度	自動削除	高度の、高度な、(高度)、(著名)、中等度の、軽度の、軽微な
分 布	注1	彌漫性、び慢性、びまん性、散在性、多発性、個在性、限局性
側 性	注2	左側、右側、両、左、右、両側
並 列	対話処理	および、及び、及、ならびに、並びに並、と、兼、「・」、「-」
病 期	括弧処理	急・慢性、新鮮・陳旧性、早期・末期
重 複	病名扱い	重複癌、多重癌、多発癌、三重癌、異時重複癌
手 術	括弧処理	切除術後、摘出術後、摘除術後、形成術後、術後、術後再発
置 換	病名扱い	大動脈置換術後、人工血管置換、僧帽弁置換、人工弁置換
人工臓器	病名扱い	ペースメーカー装着、人工心臓装着、調律器装着、蘇生器装着
放射療法	括弧処理	放射線治療後、放治後、放治後の状態、部分照射後、照射後
化学療法	括弧処理	化療後、化療後の状態、化療後再発、治療後、治療急性転化
免疫療法	括弧処理	免疫療法後、免疫療法後再発
移 植	括弧処理	骨髄移植後、骨髄移植、移植術後
合併治療	括弧処理	放射線化療後、術後化療、術後放治

注1: 多発性硬化症等を配慮に入れる。 注2: 左室肥大・右胸心等を配慮に入れる。
*: 状態記述と病名記述(例えば、「放射線性肺炎」等)を区別。

Table 4 Subset of the Concept 'Brain Infarction' (Examples)

右視床梗塞・右小脳梗塞・右側頭葉出血性梗塞・右大脳梗塞・右大脳半球梗塞・右脳梗塞・急性脳梗塞・旧脳梗塞・橋の梗塞・橋梗塞・橋底梗塞・橋底部梗塞・(局)橋梗塞・(局)脳梗塞・広範脳梗塞・出血性脳梗塞・左後頭葉出血性梗塞・左大脳梗塞・左内包梗塞・左脳梗塞・左被殻梗塞・視床梗塞・小型脳梗塞・小脳梗塞・小脳梗塞部・小脳出血性梗塞・小脳陳旧性梗塞・新旧脳梗塞・新鮮脳梗塞・新鮮脳梗塞症・前頭葉梗塞・巣状脳梗塞・側頭葉梗塞・多発性出血性脳梗塞・多発性陳旧性脳梗塞・多発性脳梗塞^{*}・多発性脳梗塞・多発性脳梗塞症・多発性脳硬塞・多発陳旧性脳梗塞・多発脳梗塞・大脳橋梗塞・大脳梗塞・大脳出血性梗塞・大脳小梗塞・大脳陳旧性梗塞・陳旧性小脳梗塞・中脳梗塞・陳旧性出血性脳梗塞・脳梗塞疑・陳旧性多発性脳梗塞・陳旧性多発脳梗塞・陳旧性脳梗塞・陳旧性脳梗塞症・陳旧性脳硬塞・陳旧脳梗塞・内包梗塞・脳出血性梗塞・脳幹梗塞・脳幹部梗塞・脳橋部梗塞・脳梗塞・脳梗塞開頭・脳梗塞後遺症・脳梗塞症・脳梗塞巣・脳硬塞・脳出血梗塞・脳出血性梗塞・脳小梗塞・脳小梗塞巣・脳小梗塞多発・脳巣状梗塞・脳多発梗塞・脳多発性梗塞・脳内陳旧性小梗塞巣・脳皮質梗塞・脳微小梗塞・被殻梗塞・微小脳梗塞・微小脳硬塞・貧血性脳梗塞・両側脳梗塞

多発性脳梗塞^{*}のように誤植と確信できるものは、「語句辞書」に含めて吸収した。

検索方法

検索の方法は、大別して、(1)補助検索を併用を前提した直接検索する方法、(2)コード化データベースを使用する方法(本システムドキュメントの提供を担当)、(3)2字文字コード和参照または3字文字コード和参照して検索時間を節約する方法、(4)シソーラス(表4参照)を用いる方法等がある。どの方法が効率が良いかは、検索目的により様々で、目的により選択する。組み合わせて使用することもある。

考察

検索方法の選択、組み合わせ使用、ならびに、検索論理・手順に効率の良い一定の方式はない。そこで、いろいろな資源を用意しておいて、検索課題に合わせて、最適の方法をみつけざるを得ないが、多くの検索で、コード化による検索で拾えなかった症例の拾い上げに成功している。

むすび

今回は、用意されている検索方法をまず紹介し、簡条書文章情報から目的の概念を検索する方法を論議する。そして、効率的利用方法について論及する。

文 献

- Kimura, S & Baba, K: Frequency tabulation of pathological findings more than 23,000 autopsy cases of annual collection through Japan. MUG QUARTERY 126-133, 1979
- Baba, K & Aizawa, S: Nationwide autopsy registration over 30 years. Autopsy in Epidemiology and Medical Research Ed. E Riboli & M Delendi, Lyon International Agency for Research on Cancer pp. 235-244, IARC 1991
- Urano, Y & Baba, K et al: Annual of pathological autopsy cases in Japan --- Computerization of autopsy data from 1974 to 1979 and their statistical study. Acta Pathol Jpn 19: 23-46, 1982.
- 馬場謙介: 剖検輯報データベース --- 集計のための作業ファイルについて. 病理と臨床6: 103-110, 1988
- 馬場謙介: %XRead スモールコンセプト (Extended Read Small Concept (%XR)). Mumps 17: 45-51, 1991 (Japanese with English Abstract)
- 馬場謙介, 藍沢茂雄, 浦野順文: 日本病理剖検輯報症例の「疾病級・級年次・年齢階級・性別頻度」. 厚生省特定疾患難病の疫学研究班昭和62年度業績報告書. 148-149, 1987.
- 馬場謙介: 日本病理剖検輯報に掲載された川崎病ならびにその疫学研究に必要な統計母数. 厚生省特定疾患難病の疫学研究班昭和57年度業績報告書. 180-198, 1982.
- 藍沢茂雄, 浦野順文, 馬場謙介: 剖検輯報によるSLE症例検索. 厚生省特定疾患難病の疫学研究班昭和57年度業績報告書. 176-179, 1982.
- 藍沢茂雄, 浦野順文, 馬場謙介: 剖検輯報によるSLE症例検索(2). 厚生省特定疾患難病の疫学研究班昭和58年度業績報告書. 271-277, 1983.
- 浦野順文, 藍沢茂雄, 馬場謙介ら: 剖検からみた老年化傾向と難病. 厚生省特定疾患難病の疫学研究班昭和58年度業績報告書. 251-262, 1983.
- Baba, K: Indications of increase of occupational pleural mesothelioma in Japan. 5: 3-15, 1983.
- 馬場謙介, 実藤隼人, 吉村健清: 日本病理剖検輯報のがん疫学への活用. 職業がん --- 疫学的アプローチ. 篠原出版, 倉恒国徳編 175-190, 1984.
- 馬場謙介: 脳卒中病型の推移に関する臨床・疫学的研究. 厚生省循環器病委託研究費による研究報告書集(藤島班) 197, 1991.
- 清水弘之, 馬場謙介ら: 難病疫学における日本病理剖検輯報の疫学的利用について. 厚生省特定疾患の疫学研究班(柳川班)平成3年度研究業績集 132-135, 1992.
- 馬場謙介: 日本病理剖検輯報の記述情報のデータ化と期待される効果. 厚生省特定疾患の疫学研究班(柳川班)平成3年度研究業績集. 141-144, 1992.
- 藍沢茂, 清水弘之ら: 日本剖検輯報を用いた難病悪性腫瘍合併率の検討. 厚生省特定疾患の疫学研究班(柳川班)平成3年度研究業績集. 1992.
- 稲葉裕, 馬場謙介ら: 剖検輯報における潜伏胃癌の検討. 厚生省特定疾患の疫学研究班(久光班)平成4年度研究業績集. 1994
- Baba, K: Enhancement of OCR data of the autopsy annual (Bouken Shuho). Mumps 18(Suppl): 134-135, 1991.

新開発環境 Easy-Tool の開発

大楠陽一、三澤哲夫、古田土義裕、小西千恵子、高橋政志、平野健一

東海大学医学部医学情報部

1 目的

東海大学医学部では、1989年よりM言語(MUMPS)による情報システムの開発をおこなってきた。これまでは住友電工のワークベンチを使用して、ベースパターンの自動コーディングを行い、内部処理や出力処理についてのオンコーディングを追加してプログラムを開発してきた。この方法は、ワープロ感覚で入力画面を作成でき、画面やコーディング標準化にも役立ってきた。

今回、システム開発の一層の効率化と標準化、実行システムの高速化を進めるために、Easy-Toolと名付けた新しい開発・実行環境を開発したので報告する。

2 Easy-Toolの概要

Easy-Toolは、Easy-PatternとEasy-Toolから構成されている。Easy-Patternは、お手本のプログラム群である。洋服に例えるとイージーオーダー服である。イージーオーダー服では、体のサイズに合ったものを選んだ後で、腕や足の長さに合わせて調整を可能している。この微調整用のコーディングに使うのがEasy-Partsである。

ウィンドウ用エミュレータとして、従来からのMWE80(住友電工)を使用しているが、エミュレータの持つ多くの機能の内、ウィンドウのオープンとクローズ機能に限定して使用しており、その他はISO標準言語の範囲で作成している。このため、他のエミュレータやGUIへの移植は簡単である。

2.1 Easy-Pattern

Easy-Patternには、主画面とサブウィンドウ画面の標準プログラムが用意されている。各プログラムには、変更不可セグメントと、追加変更および設定セグメント、自動コーディングセグメントがある。変更不可セグメントでは、メインフロー制御、ウィンドウ番号の自動管理、画面表示、編集入力、入力チェックなどが行われる。初期値設定では、画面タイトル、ウィンドウサイズ、ファンクションキー表示内容などを設定する。各入力フィールドの項目名、画面上の位置と色、入力フィールドの長さ、また必要に応じて、入力前後の処理セグメント名、ヘルプ処理セグメント名、入力フィールド前後のガイダンスは、一行のプログラム内テキストとして設定する。入力チェックの内、パターンチェック式、範囲チェック式、

間接実行によるチェック式、外部ルーチンによるチェックルーチン名は、項目名ごとにグローバルテーブルに登録され共有化されている。従って、この初期値と入力パラメータ行の設定だけで、実行可能なプロトタイプが完成する。入力前後の特殊処理、ファンクションキーとヘルプキーなどが押されたときの処理は追加コーディング可能となっている。画面制御（消去、カーソルアドレッシング、文字飾りなど）とプリンター制御（文字サイズ、改行巾、文字飾りなど）に関しては、その機能に決められたラベルのみをコーディングしておき、Easy-Partsの%TCRTCを実行し、機種を選択するとESCシケンスが自動コーディングされる。

2.2 Easy-Parts

部品の集まりであり、特殊処理、追加コーディングのために、次の外部プログラムを用意している。

(1) 入力用及びウインドウ制御用 (DO呼出)

これらはEasy-Patternの変更不可セグメントで使われており、通常のプログラマは必要としない。

```
%ZJINS   編集入力処理 (一行用)
%ZJINM           //      (複数行用)
%ZFKEY   ファンクション表示処理
%ZSELPG   選択入力処理 (ページング方式)
%ZSELRL           //      (スクロール方式)
%ZWOPN   ウインドウオープン処理 (ウインドウ番号の自動管理を含む)
%ZWCLS   ウインドウクローズ処理 (           //           )
```

(2) 日付関係ユーティリティ (外部関数)

```
$$^%ZDNOW(MOD)      システム日付
$$^%ZDTOM(X,MOD)    X日の翌日
$$^%ZDENDM(X,MOD)   X日の月末日
$$^%ZDAFTD(X,Y,MOD) X日からY日後の日付
$$^%ZDAFTM(X,Y,MOD) X日からY月後の日付
$$^%ZDCNT(X,Y)      X日からY日までの日数
$$^%ZDCHK(X,MOD)    省略日付を正式日付に変換 (一般用)
$$^%ZDCHKF(X,MOD)   //      (先付け日付用)
$$^%ZDCHKP(X,MOD)   //      (前付け日付用)
```

*MOD="S" YYYYMMDD (Standard)

"P" YYYY.MM.DD (Pointed)

"J" GYY.MM.DD (Japanese)

(3) 時刻関係ユーティリティ (外部関数)

\$\$^%ZTNOW(MOD) システム時刻
\$\$^%ZTCNVH(X, MOD) \$Hタイプからの変換
\$\$^%ZTCHK(X, MOD) 省略時刻を正式時刻に変換
*MOD="AP" AM HH:MM または PM HH:MM
"24" HH:MM (24-hour clock type)

(4) 年齢・曜日計算ユーティリティ (外部関数)

\$\$^%ZAGE(X, Y, MOD) 誕生日(X)と基準日(Y)から年齢を返す。
*MOD="Y" 年齢、"M" 月数
\$\$^%ZYOBI(X, MOD) X日の曜日と平日・休日を返す。
*MOD="N" 数値(1~7)、"E" 英語(SUN~SAT)、
"J" 日本語(日~土)、"H" 平日(W)・休日(H)

(5) 全角・半角変換ユーティリティ (外部関数)

\$\$^%ZCNVH(X) 半角変換 テーブル^%ZTDZHが必要
\$\$^%ZCNVZ(X) 全角変換 テーブル^%ZTDHZが必要

(6) ローマ字カナ変換ユーティリティ (外部関数)

\$\$^%ZROMA(X) ローマ字をカナ文字に変換

(7) 数学関数 (外部関数)

\$\$^%ZSINR(X) sin(ラジアン)
\$\$^%ZSIND(X) sin(度)
\$\$^%ZCOSR(X) cos(ラジアン)
\$\$^%ZCOSD(X) cos(度)
\$\$^%ZATAN(X) arctan(ラジアン)
\$\$^%ZEXP(X) exp(X)
\$\$^%ZNEXP(X) exp(-X)
\$\$^%ZEXP10(X) 10のX乗
\$\$^%ZLOG10(X) 常用対数
\$\$^%ZLOGN(X) 自然対数
\$\$^%ZROOT(X) 平方根

(8) その他

\$\$^%ZBCNT(X) 文字列のバイト数を返す。
\$\$^%ZFORM3(X, Y, Z) X 数値
Y 1=3桁ごとカンマ付き、0=カンマ無し
Z 丸め位置 (100 0.01など: default=1)
\$\$^%ZKHENO(X) シノニム文字化 (ツ'とズ'を同義語化など)
^%TCRTC 端末・プリンターのESCシーケンス自動埋め込み

~%ZPARAD	常用ローカル変数の設定(画面用ESCシーケンスなど)
~%ZPOPIN	ローカルプリンターへの切替え
~%ZPCLS	画面表示への切り戻し

3 結果と検討

一つの画面用のプログラムが、一本に納まり、プログラムが非常にコンパクトで、管理がしやすくなっている。東海大学医学部の予約システムのメイン画面の例では、ワークベンチによって作成されたプログラムは4本で、ファンクションキーとヘルプ処理用などのグローバルが作成され、4本プログラムのルーチンサイズの合計は約50KBであるが、Easy-Toolによって作成されたプログラムは一本でルーチンサイズは約20KBであった。

また、プログラムスワップとディスクアクセスが少ないため、実行スピードが高速化された。このため、端末のLAN直結化も全面的な効果として発揮されることがわかった。従来のシステム(端末はRS232Cの9600ボー接続)ではメイン画面からヘルプウインドウの表示に数秒要したが、新システム(端末はLANボードにより接続)では、ほぼ瞬間的に表示されるようになった。

今後機会があれば、Easy-Toolを他のエミュレータやWindows対応にもしていきたいと考えている。また、Share-ware化により、M言語の普及に貢献できればと考えている。

【参考文献】

1. 平野健一他：大規模な病院情報システムの構築、第17回日本MUMPS学術大会予稿集、32、1991.
2. 大櫛陽一：病院情報システムから地域医療システムまで—東海大学医学部・医学情報部—、医療とコンピュータ、5(3)、1-6、1992.
3. 高橋政志他：東海大学病院における予約システムの開発について、病院管理、30(1)、124、1993.

経皮的冠動脈拡張術(PTCA)施行症例データベースに使用した データ等の読込に関する Small Concepts

馬場 謙介
Kensuke Baba

国立埼玉病院臨床研究部
Department of Clinical Research, Saitama National Hospital

はじめに

外部関数(いわゆる\$\$関数)が装備されるようになったので、それまでのスモールコンセプトを外部関数の形に書き換えた。また、新規に作った外部関数型のスモールコンセプトを幾つかのシステムに導入した。経皮的冠動脈拡張術(PTCA)症例データベースに導入した後多少の改良を加え、現在、安定した動作が確認されたので報告する。

\$\$0^*XRead

このスモールコンセプトは、READ命令相当の'賢くタフな'一行エディターである。旧版にない機能の中で先ず、(1)括弧の点検機能、(2)引用符の点検機能を上げたい。この機能は、もともとルーチンエディタ用の一行エディタに装備したものであるが、この機能をこのスモールコンセプトに付加した。このことにより、検索の論理設計時にこの機能が使えるようになった。その効用を例を上げて解説する。このスモールコンセプトは、その内部で以下に解説する \$\$0^*XC, \$\$0^*XTR, \$\$NEC^*XB をサブ関数として使っている。但し、これらのサブ関数は、独立または他の関数のサブ関数として使用できるように設計してある。

\$\$0^*XCall

一度入力したデータのパターンを呼び出して、(必要なら編集して)再利用することを可能にした。呼出順序のモードは、履歴順とアイウエオ順とがある。初期モードの指定ができ、操作中にモードの変更もできる。格納グローバルに名前が指定できるので、入力項目別に名前を指定付けて使用している。この機能は、患者住所、臨床経過等の記述情報の漢字変換の便を期待して設けたが、定型診断の入力にも有効であった。検索論理の記述の'試行錯誤'の過程で、このスモールコンセプトは最も威力を発揮してくれた。このスモール

コンセプトの導入により、目的の検索の本番ランまでの所要時間が大幅に短縮できた。

\$\$NEC^%XBoard

複数の研究者が複数の別装置で別個にデータ検索する事態になった。このスモールコンセプトによって、任意のキー設定に対処できるようになった。代表としてこの項の見出しに、\$\$NEC^%XB を使ったが、\$\$NEC93^%XB, \$\$VT52^%XB, \$\$IBM55^%XB 等を用意して、この事態に対処している。尚、\$\$0^%XR, \$\$0^%XC 等、外部関数型のスモールコンセプトは、ショートカットキーによるキー操作を基本として設計してあって、これを上記の \$\$XB関数で装置固有のキー定義をショートカットキー相当に変換してくれる。

\$\$0^%XTRans

\$\$0^%XTRは、\$\$0^%XRのサブ関数として、カタ仮名→ひら仮名、ひら仮名→カタ仮名、カタ仮名→カタカナ、カタカナ→カタ仮名、ABC→ABC、ABC→ABC 等の変換をキー操作時実行するようにした。更に、例えば、「んんY」のいずれが押されても「Y」に変換するモードを全キーに対して設けた。また逆に、英数字をカタカナに変換するモードも設けた。(但し、英数字キーとカタカナキーは必ずしも対応していないし、装備により差もあるので完璧でないことを承知の上使用されたい)。変換モードを入力項目にタイトに定義することができるので、入力者を項目毎の仮名キー・漢字変換キーの操作の煩わしさから解放した(操作時に他変換の指示可)。

考 察

これらのスモールコンセプトによって、データ入力に対するシステムの質は向上した。また、検索論理の作成の手助けに大いになった。M言語でもウィンドウが開設できるようになると、入力の操作性は大いに向上すると期待される。それは、有難いことではあるが、ここで述べた括弧や引用符の点検機能、呼出機能のブラッシュアップにも傾注していかなければならないのではなからうか。

文 献

- 馬場謙介: %XRead スモールコンセプト (Extended Read Small Concept (%XR)). Mumps 17: 45-51, 1991
(Japanese with English Abstract)
- 馬場謙介: コーディングセンス. 中級M言語(中山書店/印刷中)
- 馬場謙介ら: 予後解析用汎用データ処理システム --- 経皮的冠動脈拡張術予後解析. IRYO 43: 856, 1989, Abstract only, in Japanese

入院病歴管理システム・経皮的冠動脈拡張術症例データベースに応用した 呼出機能を持つ編集可能疑似READ命令

Editable PseudoReadCommand with Data Call Capability

馬場謙介

Kensuke Baba

国立埼玉病院臨床研究部

Department of Clinical Research, Saitama National Hospital

はじめに

電子計算機の'READ'機能は、いわば、ルーチンの顔である。殆どのオペレーティングシステムは、入力時に(↑キー等の操作で)履歴呼出機能が使えるように設計されている。しかし、これと同様の機能を組み込んだルーチンは数少ない。そこで、履歴呼出・アイウエオ順呼出の機能を持ち且つ編集可能な疑似READ命令に相当する Small Concept を自作し、実際のルーチンに組み込んだところ効果があったので紹介する。

方 法

PC-9801ns(2Mバイト)(日電)上のSP-MUMPS version5.0 (住電)で基本バージョン開発し、PC-9801-RX(日電), PC-9801-DX(日電), PC-98-Bp(日電)等に移植した。このスモールコンセプト(^%XC)は、(WordStar仕様に準ずる)ショーカットキーによるキー操作を保証している。従って、機種を問わずショートカットキーで操作できる。カーソルキーに慣れたオペレータを念頭に、エンジン内のサブ関数の\$\$device^%XBOARDによって、任意のdeviceのカーソルキー操作定義に従って操作できるように設計した。

履歴順呼出

履歴順呼出機能付きの入力は、履歴呼出括弧で区切られたフィールドの中で行う。履歴順呼出を示すプロンプト括弧は、[]である。履歴順呼出の操作は簡単で、[CONTROL[W]]で前回、[CONTROL[X]]で次回入力の値が表示される。求めるデータが表示されたら、リターンキーで疑似READ命令は完結し、その値を返す。また、その直前にそのデータを最新のデータとして履歴順格納グローバルに格納する。但し、そのデータの値が、前回履歴のデータと一致した場合、履歴順格納グローバルを更新しない。疑似READ命令が完結する前に文字列を編集キー(表1)等で編集できる。指定した最大記憶領域(max)を越えようとすると、最も古いデータは捨てられる。リング格納は、定石通り n#max を節値として実施した(ここで、nは通算入力回数)。

アイウエオ順呼出

アイウエオ順呼出機能付きの入力は、履歴呼出括弧で区切られたフィールドの中で行う。アイウエオ順呼出を示すプロンプト括弧は、< >である。アイウエオ順呼出の操作も簡単で、[CONTROL[W]]でより低値の既存データ、[CONTROL[X]]でより高値の既存データが表示される。目的のデータが表示されたら、リターンキーで疑似READ命令は完結し、そのデータがアイウエオ順格納グローバルに格納される。このモードでも、疑似READ命令が完結していなければ(リターンキーを押す前であれば)、文字列を編集キー(下表)等で編集できる。

相互乗り入れ

アイウエオ順呼出モードで入力したデータは、履歴順格納グローバルにも格納される。逆に、履歴順呼出モードで入力したデータは、アイウエオ順格納グローバルにも格納される。また、アイウエオ順呼出モードの状態から履歴順呼出モードの状態にモード状態の変更することもできる。

Small Concept の外部仕様

このスモールコンセプトは、外部関数(\$\$関数)の形で下の如く定義され。

```
$$0^%XC(String,Mode,Name,Max,Left,Right)
```

ここで、各パラメータは表2の通りである。

入院病歴管理システムにおける応用

入院病歴管理システムでは、(1)画面中央に表示する症例の指定、(2)印刷開始位置の指定、特に効果(3)印刷終了の指定、(4)印刷リストのタイトルの入力(「督促状」!再督促!」「!!!!再々督促!!!!」「胃癌症例検索リスト」「東二病棟入院カルテ一覧」等はインストール時に入力済み)、(5)印刷リストの発送先の等の入力(医師全員を入力済み)のために使用した。特に、(4)(5)では漢字入力を伴うので重宝されている。

経皮的冠動脈拡張術症例データベースにおける応用

経皮的冠動脈拡張術症例データベースの入力では、(複雑であるがパターンが決った)定型データの入力が増える。この機能の導入によって、入力が容易になりかつ小さいミスも防止できるようになった。また、検索時に一度使った検索論理をそのままあるいは編集で変更して使えるので、(トライアンドエラーが進めることが多い)検索の作業効率を

著しく向上させた。

考 察

オペレータが快適な気分になる程、作業効率を向上させることができた。一応の成功を得たと考えるが、このスモールコンセプトを土台に辞書型の呼出に発展させることが次の課題である。プロンプトに二組みの括弧を使用した理由は、この仕様をそのまま一行編集に持ち込んで、一行編集行上に括弧で開く可変長小窓を設定して、ここで履歴呼出(・辞書呼出)をしたいからである。

Table 1 Definition of Short Cut Keys

Short Cut Key	Definition
[CONTROL[A]]	Cursor Back-ward
[CONTROL[D]]	Cursor Fore-ward
[CONTROL[W]]	Cursor Up-ward
[CONTROL[X]]	Cursor Down-ward
[CONTROL[V]]	Delete Character
[CONTROL[H]]	Back Spacing
[CONTROL[R]]	Insert Mode Toggle
[CONTROL[Y]]	Data Delete

Table 2

Parameter	Meaning
<i>String</i>	Initial String
<i>Mode</i>	Initial Mode [*]
<i>Name</i>	File Name
<i>Max</i>	Maximum of Histry
<i>Left</i>	Left Prompt
<i>Right</i>	Right Prompt

*: 0=Histry, 1=Alphabetic

文 献

馬場謙介ら:コーディングセンス. 中級M言語(中山書店/印刷中)

馬場謙介: %XRead スモールコンセプト (Extended Read Small Concept (%XR)). Mumps 17: 45-51, 1991
(Japanese with English Abstract)

MS-WindowsとDTM環境におけるGUI開発方法の比較

Comparison of GUI Development Way
in MS-Windows and DTM Environment

波川 智亮、天野 満、嶋 芳成

Namikawa, Chiaki Amano, Mitsuru Shima, Yoshinari

日本ダイナシステム株式会社

Japan DynaSystems Inc.

1、目的

最近、Macintosh は日本国内での市場を広め、MS-DOS の世界でも MS-Windows が普及している。エンドユーザーは、視覚的にわかりやすく操作性の共通化されたシステム、マウスなどをサポートしている扱いやすいシステムを求めている。M言語は GUI のサポートがまだ遅れており、GUI の機能を充実したシステム開発は難しいと一般に思われている。

今回、DTMのウィンドウズオプションである DT-Windows と、Visual Basic および WinPlus という Windows アプリケーションを組み合わせてサンプルシステムを開発した。そして、開発手段の違いによる GUI システムの開発の容易さや実行処理速度などについて比較検討したので、ここに報告する。

2、方法

データベースのM言語に、DTM-PC-1 Ver.4.5J とそのウィンドウズオプションである DT-Windows Ver. 2.1 を組み合わせた。

マンマシンインターフェースの手段には、以下の5種類を用意した。

- 1) DTM の従来の文字ベースのマンマシンインターフェース (CUI: DOSウィンドウを使用)
- 2) DT-Windows のオリジナルの構文によるウィンドウの作成
- 3) MDC による M Windowing API の構文を用いたウィンドウの作成
- 4) Windows アプリケーションに、Microsoft Visual Basic Ver. 2.0 を利用
- 5) Windows アプリケーションに、WinPlus Ver. J2.52 を利用

OS 環境は、MS-DOS Ver. 5.02J/V、MS-Windows Ver. 3.1、ハードウェアは、FMV-466D (80486DX2/66MHz、RAM12MByte) である。

サンプルデータとして、10万件のデータを作成し利用した。

3、結果および考察

GUI に関わる実行速度は、MS-DOS の CUI に比較するとかなり劣る。遅くなる理由は Windows 自身のオーバーヘッドが大きいことと、Windows と DTM の間のデータ交換の速度が遅いことによる。しかし、大量のデータを扱った結果としては、我慢の範囲内と思われた。

近い将来、GUI がマンマシンインターフェースの基本となることは間違いない。強力なデータベース/ネットワークのエンジンであるM言語は GUI のパラダイムでもその価値は損なわれるものではないが、GUI アプリケーションが容易に開発できるということは、歓迎すべき進歩である。

今回開発したシステムの内容の詳細、実行処理速度等については、当日紹介する。

M環境と外部世界との融合の試み

Attempts to integrate M environments and outer worlds

佐藤比呂志

H.Sato

日本デジタルイクイップメント(株)
西日本第一統合システム部

Digital Equipment Corporation Japan
Enterprise Integration Center

〒530 大阪市中中之島2丁目2番2号
ニチメンビル

TEL(06)-222-9211 FAX(06)-222-9408

Software environments that are provided by M language are strong enough to establish most working systems by them own. We have been making efforts to integrate M environments and required capabilities step by step. It can be said that usability and practicability of M environments are mainly due to the efforts.

M environments, however, are significantly behind with recent rapid progress of computer environments.

In order for M to continue to develop, conventional efforts for integrating anything into M environment are not appropriate. It may be required to make efforts to have deeper relationship with worlds other than M and to coexist with them.

The following introduces some of the attempts which are made in order to link M environments and outside worlds together by our company.

要約

M言語が提供するソフトウェア開発環境は、それだけで大抵の実用システムを作成できるほど強力なものです。いままでM環境は、必要とされる機能を順次自分自身の環境に取り込む努力を続けてきました。M言語の使いやすさ、実用性は、これらの努力によるところが大きいと思います。

しかしながら、昨今のコンピュータ環境の急激な進歩に対して、M環境の対応の遅れが目立ってきました。

Mが今後とも発展するためには、従来の様に何でもM環境に取り込もうとするのではなく、M以外の世界との関係を深め、共存するための努力が求められてきているのではないのでしょうか？

その様な観点から、現在弊社で試みているM環境と外部世界とを結び付けるいくつかの試みをご紹介します。

キーワード: M環境 M言語 外部世界

DSM API for Windows

高性能 PC の普及に伴い、MS-WINDOWS インタフェースを使用したアプリケーションを使用したい、開発したいという要求が強くなってきました。

この要求に対するひとつのアプローチの方法として、PC 上に M 言語を載せ、さらに MWAPI (M Windowing Application Programming Interface) を実装するということが考えられます。しかし、残念ながら、DSM を既存の Windows 3.1 の環境に実装するという計画はありません。(DSM が動作するのは、Windows 4.0 以降の環境上になります。)

そこで、既存の DSM 環境に PC の MS-WINDOWS 環境からアクセスするためのネットワーク・ツールを開発しました。そのツール名は、DSM API for Windows です。

このツールを使用して PC 側からコマンド実行、グローバル、ローカル変数へのアクセス、ルーチンの実行、M 関数の実行を DSM サーバーに依頼することが可能になります。

このツールと PC 側のツール (Visual Basic 等) を組み合わせることにより、MS-WINDOWS の GUI を駆使した DSM アプリケーションを作成することができます。

このツールの利点

- 低コストで GUI M アプリケーションを作成できる。
- 習得が簡単
- M 言語をご存知の方であれば、このツールが提供する API を理解するのは、とても簡単です。
- DSM のほとんどの機能を利用することができる。
- クライアント・サーバー・コンピューティング

このツールの弱点

- データの引き渡し

この問題は、このツールに限らず、M 環境と他環境を結合する時にいつも問題となります。

M 言語と他の言語でデータを引き渡す場合問題となるのは、M 言語の変数構造が他言語、他環境には理解できないという点です。この制限のため、引き渡せるデータは、単純な文字列や数値データに限定され、M の高度な構造をもった複雑なデータを引き渡すことができません。

- PC 側に M 以外のツールが必要

M 言語以外に Visual Basic 等の習得が必要になります。

SQL/DSM

この製品を使用することにより、PC 上の様々なツール (MS-Access MS-EXCEL, Visual Basic 等) から他のリレーショナル・データベースと全く同じ様に DSM のグローバル変数にアクセスすることができます。このことは、M を全く知らないエンドユーザーが、DSM グローバルを様々な角度から取り出すことができることを意味し、DSM 環境で、EUC (End User Computing) の道がひらけます。

このツールの利点

- 様々な PC ツールから DSM データベースを利用可能
- EUC が可能
- クライアント・サーバー・コンピューティング

このツールの弱点

- M 特有の機能は使用できない。

M のことを意識せずにアクセスできるということは、逆に M の特色をだすことが難しいことを意味し、他のリレーショナル・データベースと差別化が難しくなります。

MIDDLE

(M Interface Data Definition Language Extension)

M 環境と外部世界でデータを交換する場合の現状の問題点は、スカラーデータのみでの交換に限定されており、配列やより複雑な構造をもったデータの交換ができない点です。また、他の言語に存在するデータ・タイプの区別も M 言語にはありませんので、データ・タイプを意識したデータ交換もできませんでした。

この問題を解決するためのメカニズムである MIDDLE を現在開発中です。

MIDDLE 開発の目的とゴール

- DSM と C 言語間のシームレスな相互データ交換を実現する。
- DSM と既存の C 言語で記述されたプログラムまたは、ランタイム・ライブラリとの結合を可能とする。
- データを交換するための特別なルーチンを介在させることなく実現する。
- C 言語で記述された関数と同じ様に DSM のルーチンを呼び出すことを可能にする。

MIDDLE とは

インターフェース定義言語を提供します。

- C 言語の複雑なデータ構造を M 言語のデータ構造にマッピングします。
- M ルーチンと C 言語の関数プロトタイプをマッピングします。
- 既存の外部呼び出し機構 (ECALL) と Callable interface を拡張したものです。

MIDDLE によって可能になること

DCE RPC と M 環境との結合

ORB(Object Request Broker) と M 環境との結合

標準ランタイム・ライブラリとの結合

既存の C 言語パッケージソフトウェアと M 環境の結合

おわりに

M を取り巻く外部環境は、日々刻々と変化しています。これらの変化に M は遅れることなく追従していくことが求められています。

今回紹介した試みは、それを解決するためのほんの一例にすぎませんし、すべての問題を解決するものでもありません。

今後とも M 環境と今後重要になるであろうオブジェクト・モデルに基づいた新しい外部世界とをどの様に融合すべきかなど考えていかなければならない問題がいくつかあると思います。

また、その様な融合された環境での M の強み、メリットが何であるのかをはっきり打ち出せなければいけないのではないのでしょうか。

獨協医科大学病院薬剤部 薬歴管理支援システム

木村一元¹⁾, 本田雅巳²⁾, 岩瀬利康²⁾, 斎藤昭好²⁾,

笹原 茂³⁾, 飯島一夫⁴⁾, 藤田 知⁴⁾

- 1) 獨協医科大学総合研究施設医用電子工学室、
- 2) 獨協医科大学病院薬剤部、
- 3) 株式会社リコー システムユニット事業部、
- 4) インターソフトウェア株式会社

入院調剤技術基本料業務(400点業務)を契機にコンピュータによる薬歴管理を考え、その支援システムの構築を行なった。このシステムは、6人の薬剤師によるデータの同時入力・検索ができ、さらにデータの一元管理ができるように、パソコン上で稼動するDTMマップスを用い、パソコンをイーサネット(10BASE5)にて接続し、ネットワーク構成とした。システムの開発は、マンマシンインターフェイスを考慮してメニュー方式を採用し、文字入力を極力少なくした。

1. はじめに

入院調剤技術基本料業務(400点業務)は、薬剤師による入院患者さんへの「服薬指導」と「薬歴による管理」を中心とした業務である。この業務の中で、投薬・指導記録の作成内容については、投薬歴・副作用歴・薬学的管理の内容・指導内容など多項目にわたり、詳細に記述し、正式な記録として保管することが義務付けられている。そして、これらの記録に基づいて薬剤管理と適切な服薬指導を行ない、必要に応じてその内容を関係者へ迅速にフィードバックし、疾病の治療や患者さんのクオリティオブライフ(QOL)の向上に寄与することが、この業務の主な目的である。

400点業務において、手作業での投薬・指導記録作成に費やす時間は、業務全体の中で大きな割合を占め、400点業務をより多くの患者さんに普及拡大して行く、大きな障害になっている。また患者情報の検索抽出の迅速性も大きな問題となる。そこで、投薬・指導記録作成に費やす時間の短縮を第一の目的として400点業務支援のためのコンピュータシステム導入を考えた。システムの導入に当っては、他施設の状況を調査した。これによると、発生源入力によるオーダーリングシステムを実施している施設での投薬・指導記録のコンピュータ化の報告や、調剤薬局におけるパソコンによる薬歴管理システムの報告は見られたが、これらを導入していない一般の病院薬剤部での400点業務に適合したパソコンソフトの市販品は見られなかった。

今回、パソコンによるマンマシンインターフェイスを考慮した400点業務支援システムを開発したので報告する。

2. システム構成

1) ハードウェア構成

使用したパソコンは、DOS/VマシンとNEC98で、サーバー1台（IBM PS/V 2405, i486DX2-66MHz, 340MbHDDx2）とクライアント6台（IBM PS/V 2405, i486DX2-66MHz, 340MbHDD, x5 + NEC PC-9801BX/U7, i486SX-20MHz, 120MbHDD）をイーサネット 10BASE5 にて接続しネットワークを構成した。NEC98は、学内他部門との文書ファイル、データ交換も行なう。プリンタは、レーザープリンタ2台（リコー SP8 x 2）を使用し、DOS/Vマシンの1台とNEC98とにそれぞれ接続した。

2) ソフトウェア構成

400点業務は、6人の薬剤師で行なわれており、投薬・指導記録作成が同時にできなければ、それに費やす時間を短縮することが困難である。また、データが一元管理されなければ、入力したデータの有効利用ができない。ネットワークの管理およびプログラムは、パソコン上で稼動するDTMマンプスを用いて行なった。

システム開発は、マンマシンインターフェイスを考慮してメニュー方式を採用し、文字入力を極力少なくした。すなわち、番号の入力あるいは矢印キーの移動指示で所定の項目が選択できるようにした。また、各項目に入力するデータも同様にメニュー（データ一覧）がその項目の脇に表示されるようにした。作成したグローバルファイルは、指導記録ファイルを含め18ファイルである。

初期メニューは、グランドメニューとサブメニューとに分かれており、グランドメニューを選択するとその右側にサブメニューが表示される。薬歴管理支援システムのプログラムは、大きく分けて7業務から成り立ち、各業務は、以下の仕事に分かれている。

1) 入院管理業務は、

1. 患者属性情報入力、2. 入院患者一覧表発行、3. 投薬管理記録簿発行、
4. 投薬管理記録画面表示の4項目、

2) 投薬管理業務は、

1. 処方箋・注射箋入力、2. 薬学的管理入力の2項目、

3) 服薬指導業務は、

1. 服薬指導表発行、2. 服薬指導結果入力、3. 医師への連絡表発行の3項目、

4) 集計処理業務は、

1. 実施患者一覧表発行の1項目、

5) 医薬品管理業務は、

1. 医薬品管理、2. 薬効分類管理、3. 問屋・メーカー管理、4. 剤型管理、
5. 医薬品情報FD入力の5項目、
- 6) マスタ管理業務は、
 1. 薬剤師管理、2. 病棟・病室管理、3. 診療科管理、4. 医師管理、
 5. コメント管理、6. 疾病名管理、7. 処方箋セット管理、8. 用法手技管理
 9. マスターリスト発行の9項目、
- 7) システム管理業務は、
 1. 定数設定、2. ファイル更新、3. オペレータ登録の3項目

3. 結果・考察

投薬・指導記録作成に費やす時間の短縮を第一の目的として400点業務支援のためのパソコンネットワークシステムを開発し、本稼動して3カ月以上経過した。当初は、手書きからキー操作への移行と不慣れな点が作業時間と疲労とを増大させたが、データの蓄積と共に記録簿の検索、過去の患者検索、各種データの集計の時間の短縮、各記録簿の読み易さ、データ入力の慣れなど各要因が相乗して、投薬・指導記録の作成時間の短縮となり、400点業務の実施病棟数を増すことができた。

西川町総合保健医療福祉サービスエリアにおける
地域保健・医療・福祉行政総合支援システムの構築

*
土田 伸・小川 一博・奥山 純二
(西川町役場保健福祉課)

1. はじめに

(1)西川町の概要

西川町は、山形県のほぼ中央に位置し、磐梯朝日国立公園の朝日連峰や月山などの景観あふれた山岳に囲まれ、東西24km、南北33kmに広がる、総面積 393.231km²と県下第4位の面積を有している。

町面積の95%が山地であり、標高は最低 145m、最高 1,980m(月山)で50余りの集落は寒河江川沿いとその支流の山間に形成され、標高 700mという高冷地まで生活の場となっている。それだけに四季折々の山岳修景美と豊富な山菜などがあり、まさに自然の宝庫である。

気象は、典型的な裏日本型気候で、全般的に日照時間が少ないうえに気温が低く湿潤であり、冬は全国屈指の豪雪にみまわれ寒さの厳しい積雪寒冷地帯である。そして山間部と平地部との気象に著しい差があり、自然的条件は、必ずしも恵まれているとは言えない。

西川町は、昭和29年10月 1日4村合併により誕生し、直後の昭和30年には15,250人あった人口も減少の一途をたどり、平成 6年には、8,380人(H6.4.1)とこの39年間で約半数まで減少するに至った。

また、若年層の流出、出生率の低下と相まって高齢人口が急速に増加しており、高齢化対策が急務となっている。一方交通網については、国道 112号の全線改良がなされ内陸と庄内を結ぶ交通の要所となっている。

さらに、町の道路交通については、各集落までの改良、舗装は整備率が高く、加えて除雪体制の整備とともに冬季交通も全集落まで確保されている。また、交通弱者に対しては町営路線バスの運行により地域住民の足の確保を図っている。

今後、東北横断自動車道酒田線(山形自動車道)の開通により、首都圏域及び圏内外の都市間との時間短縮が図られ、まさに高速交通ネットワーク体制へ組み込まれることになる。

前記の背景により、自然的・経済的条件の劣悪と、人口の減少と高齢化の上、昭和47年度における町民1人当たり所得が 412千円と全国比57.5%、県比74.8%、山間地特有の短命などの状況の中、次の様な施策を展開している。

- ①県民平均所得への挑戦 (追いつけ・追い越せ)
- ②地域資源の活用 (マイナス条件をプラスの条件に)
 - ・月山夏スキー
 - ・月山自然水
 - ・ふるさとクーポン
 - ・月山メノウ 等々
- ③意識の改革・発想の転換(嘆き・口説き・ボヤキの払拭)
 - ・価値感の醸成、クオリティ・ライフの実現運動
 - ・ひとづくり、西川塾の開設、町民の国内外派遣制度
 - ・職員の提案制度
 - ・婦人会一口発表会、町長との対話の集い (意識の変化)
- ④健康づくり、長寿のまち、きめ細かな福祉対策
 - ・山形県内でもトップクラスの長寿の町となった。(H3年 平均寿命 男75.7才 女81.2才)
 - ・満 100才に、100万円の敬老制度
 - ・寝たきり老人等介護者に対する激励金の支給制度
 - ・高齢者の除雪対策事業の実施
 - ・各種検診の受診者数と受診率向上 (総合及びドック受診率 66.6%)
- ⑤生活環境、教育環境の整備
 - ・生活道路の100%舗装化、早朝完全除雪体制の整備
 - ・小中学校は、統廃合しない。(小学校9校、中学校3校の校舎改築終了)
- ⑥産業基盤の整備

(2)まちづくりの方向

①5つの条件整備

- ・交通アクセス（横断道酒田線、I C町内2ヵ所、山形空港から車で30分）
- ・下水道の整備（平成5年度より着手）
- ・水資源（水資源豊富、水文化の構築）
- ・風景・景観（景観の保全、集景整備事業）
- ・情報（人的交流、人的ネットワークの強化）

②長寿社会への対応

- ・保健・医療・福祉の総合施設整備と在宅福祉の施設・ソフト面の強化
「保健・医療・福祉サービスエリアの総合施設」

（町立病院・老人保健施設・特別養護老人ホーム・ディサービス・老人住宅・保健センター）

④経済基盤の整備及び企業誘致

- ・地域特性を活かした特産物の開発と作物の奨励
- ・林道網の整備（徹底したコストダウン）
- ・広域圏域内等での良質な企業導入

⑤株式会社の設立

- ・町及び町主導で実施してきた事業の経営を改善し、収益的事業をも包括する新たな企業の設立。

⑥西川独特のリゾート地域の開発

- ・地域住民の福祉の増進に寄与する地域活性化の手段。
- ・地元資本で行う。

⑦クオリティ・ライフの実現

- ・物の豊かさから、心の豊かさに。本当の豊かさの追求。
- ・クオリティ・ライフ研究所（町民・行政への提言、学習、実践機関）

⑧その他

- ・月山湖大噴水
- ・月山湖 「水の文化館」

(3)西川町の「総合保健医療福祉サービスエリア」と「情報システムの必要性と目標」

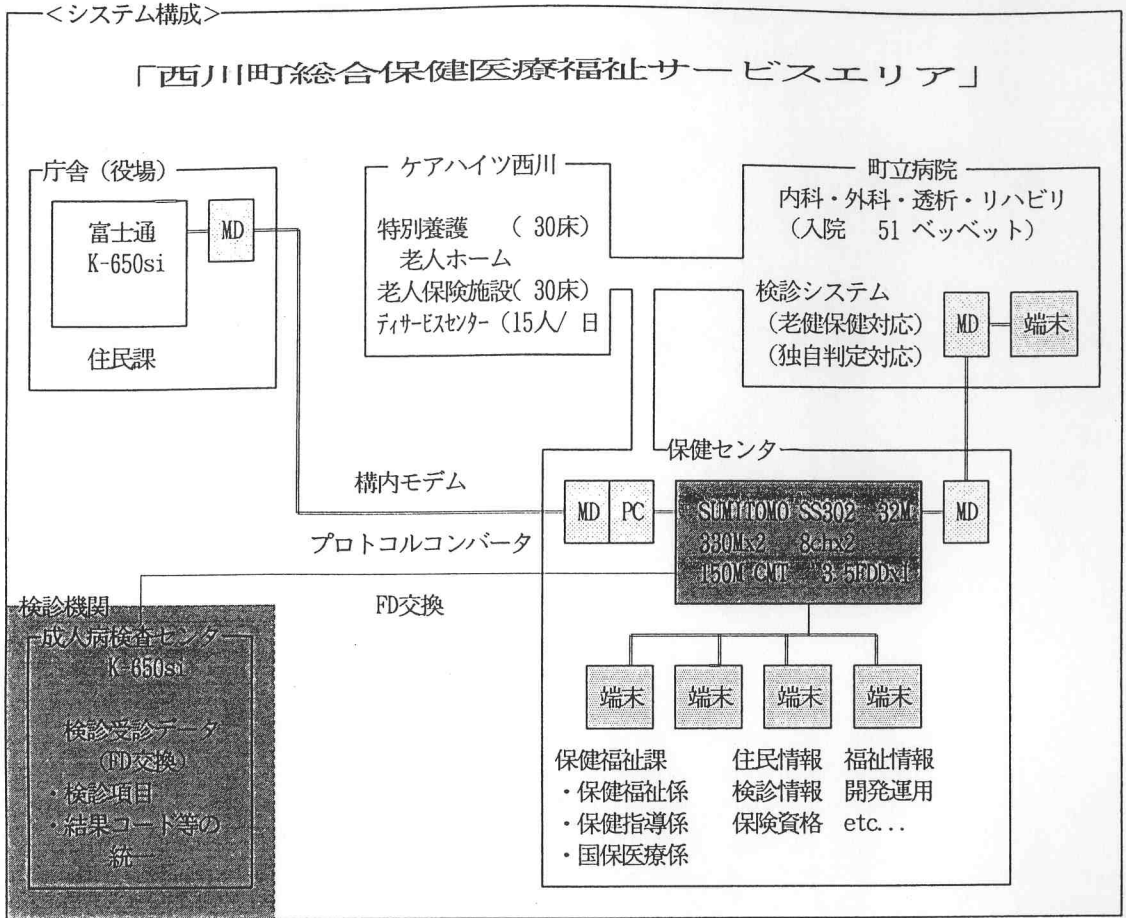
西川町では、西川町総合保健医療福祉基本計画に則り、保健・医療・福祉サービスの一元化を図るため、町立病院を中心とした総合保健医療福祉サービスエリアを設定し、町民の健康づくりを目指している。特に、高齢者サービスを重点に施設整備を行っており、老人保健福祉施設「ケアハイツ西川（特別養護老人ホーム・老人保健施設・ディサービスセンターの合築施設）」も同一エリア内に整備している。これらの保健・医療・福祉サービスエリア内に保健センターを平成5年度までの基本計画における高齢者住宅を除く各施設の完成となった。

また、保健センターは、保健・医療・福祉サービスエリアで保健福祉サービスシステムの拠点となるもので、町民一人ひとりが生涯にわたって健康づくりを推進するため、保健衛生思想の普及に努めると共に、健康教室・母子保健事業・各種予防接種・老人保健事業など地域に密着した対人福祉サービスを総合的に行い、健康で明るい町づくりを推進するものです。

このような保健・医療・福祉サービスエリアの中核的部署である保健センターでの保健・医療・福祉の行政分野における事業も多様化し、スタッフに対する負担も増大している。そのため保健・医療・福祉の一貫した継続性のあるこれら分野の情報を世帯又は、個人単位に時系列に管理し、高齢化社会に適応できる、より包括的・一元的保健・医療・福祉の行政事業を推進するため、これら事業のバックアップシステムとして、「西川町保健・医療・福祉行政支援システム（総合データバンクシステム）」を「地域医療情報化システム」を包括したシステムとして、昭和62年度に町立病院での「受診者健康管理データバンクシステム」・昭和63年度に保健分野での「健康管理データバンクシステム」を導入し、平成2年度においては、これまでのシステムを西川町保健・医療・福祉基本計画に則った保健・医療・福祉サービスエリアでの保健センターの機能に耐えうるシステムの再構築として取り組み現在に至っている。

上記のシステム化の背景を踏まえて、本システムは、庁舎（役場）で管理している住民基本情報を基に、保健センター内にメインマシンを設置し、国民健康保険被保険者資格情報・税情報・健康診断情報・医療情報（国保・退職・老人・県単の各被保険者に関する診療報酬データ等）・予防接種情報・福祉サービス情報等を管理し、対象者に対するケースマネージメントの確立を図ることを目的とする。

2. システム構成



①機器構成

・メインマシン	住友電気工業(株)	Sumi Station(メイン機)	32Mb/CMT 150Mbx1/HD 330Mbx2/FDD '3.5x1)
・無停電電源装置		2 KVA	
・構内モデム		BSC-502	x 4 (庁舎・病院間)
・プロトコルコンバータ		PC-1	x 1
・レーザープリンタ		LAS-B406E	x 3
・多機能端末		FMR系	x 5
・ターミナルプリンタ		LAS-PR355	x 1

②ソフト開発

・OS	SEI UNIX
・開発環境	U-MUMPS
・その他	基幹システムを住友システムズ(株)に委託開発。 その他、独自開発による。

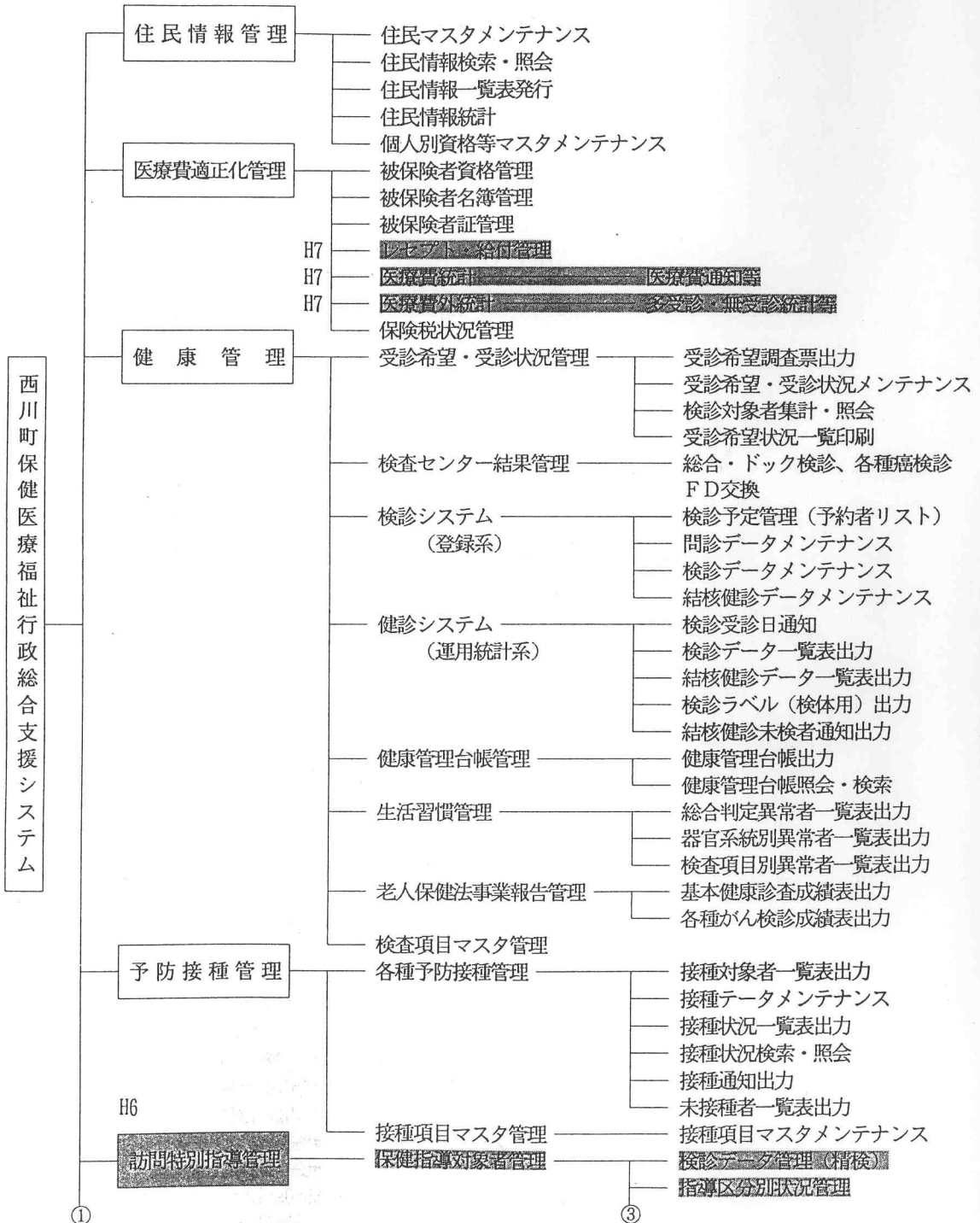
③特徴

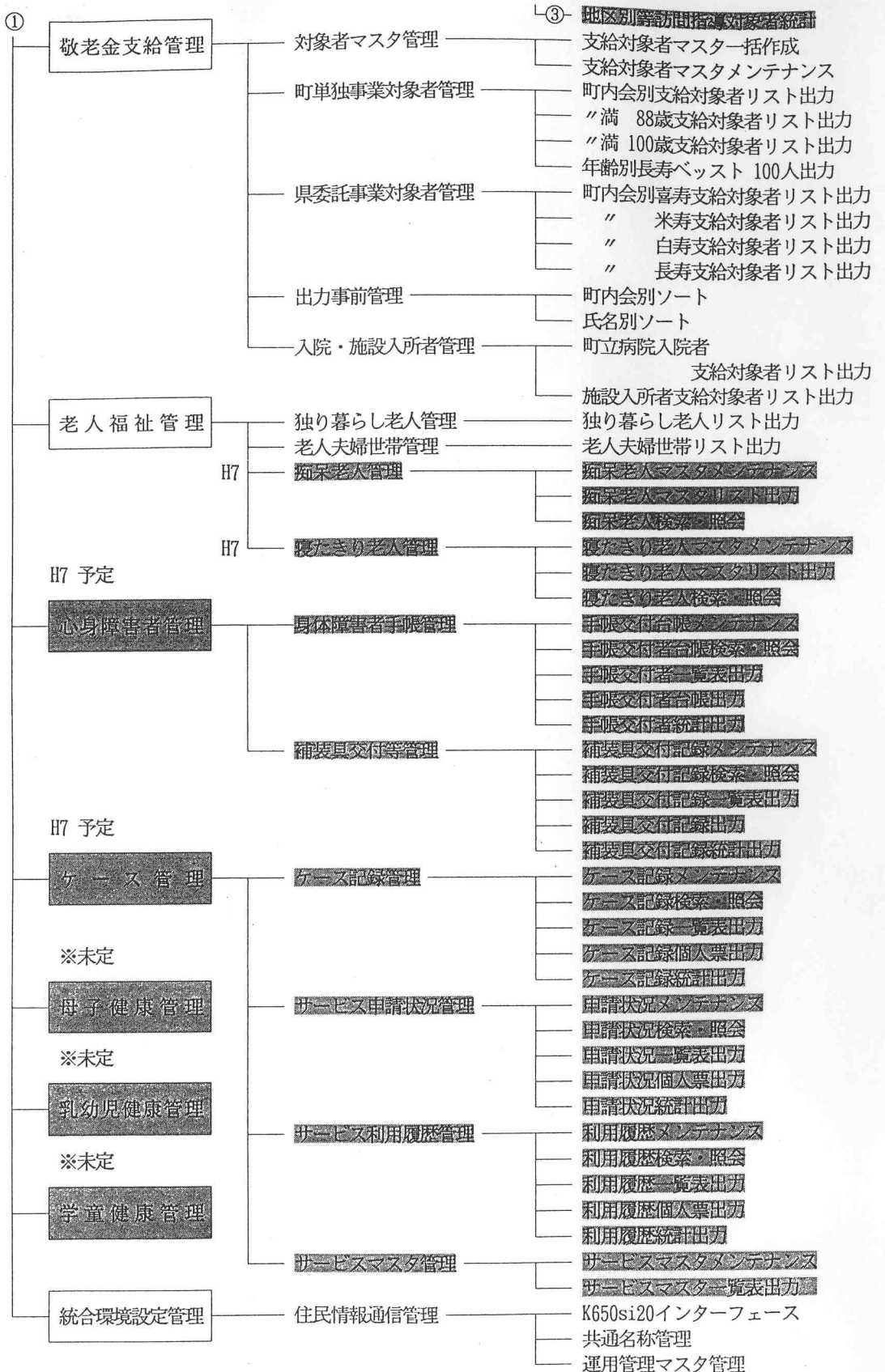
- これまで導入してきたシステムでの障害であった保健・医療・福祉のサービスを長期的に個人に対し行うために必要とされる住民情報の履歴管理を、庁舎(役場)より構内モデムとプロトコルコンバータを介して転送される住民異動情報の蓄積により、全ての履歴の保存管理を行っている。この点は、個人に対して生涯的なサ

ービスを行ううえで極めて重要とされ、同様のシステムを導入している県内の自治体で、この点の運用問題から、システム全体の運用に影響を与え、システムの運用が行えないところが多くみられる。

- ・ 住民検診の検査データは、検査機関での結果及び判定コードのコード、判定テーブルも準用し、また、町立病院の独自判定及び老人保健法による判定にも対応を行っている。いわば、検診機関の検査システムをも踏襲しているシステムである。

④機能一覧





3. 現状

現在、保健・医療・福祉のサービスにおける個人へのケースマネジメントを確立することを目標とし、そのサポートシステムとして、運用を開始しているが、今のところ作業効率の高い検診システム関連の運用が主になっている。

また、サービスエリア内においては、各部署において個人に対するケースマネジメントを基本とし、サービスの連携・的確・迅速性等を重視した対応を心掛けているが、以下の点において極めて有用となっている。

- ① サービスを行う上で、基本となる住民情報の履歴を含んだ即時的な把握が可能となった。
- ② 成人病検査センター、結核予防協会、町立病院等で行っている各種検診に対するそれぞれの機関毎の判定が、概ね標準化することが可能となり、保健指導の平準化や、時系列での管理が可能となった。

4. 今後の展望（「共に健やかに、生きいきと、そしていつまでも」）

現在、保健所法が改正され、保健・医療・福祉の連携が求められているが、西川町では、この点に一早く着目し、「共に健やかに、生きいきと、そしていつまでも」をキーワードに、これまで重点的に拠点施設の整備を行ってきた。今後、関係機関での情報の相互利用と連携を行いながら、サービスを受ける方への問題・ニーズ把握、ニーズ評価、ケア計画検討・決定、サービス陽性・手配、サービス実施・提供、サービスの観察・評価を循環して行うケースマネジメントとその業務連携、しいては、サービス体制及びシステムの内部評価を行うために、情報システムを有効に活用していきたい。そのためにも、今後、次の点において機能を付加し、生涯にわたっての住民の健康づくりに役立てることを目指す。

- | | |
|----------------|-------------------|
| ①保健指導管理システム | ⑤母子保健管理システム |
| ②老人福祉管理システム | ⑥乳幼児健康管理システム |
| ③心身障害者福祉管理システム | ⑦学童健康管理システム |
| ④ケース管理システム | ⑧統計システム（先行的機能の強化） |

更に、本システムの広域的医療圏での利用と、住民の健康管理への一層の意識の高揚、プライバシー保護の観点から、本システムの理解を得たうえで、将来直接利用者となる医療機関や関連福祉施設などの利用申請方式、及びデータの登録者である住民自らのデータ登録管理に関する登録利用申請方式を導入し、運用システム体制の整備を行う予定である。

5. さいごに

本システム構築は、厚生省主管の国民健康保険の各特別事業の補助を受けて実施おり、また、基幹システムの開発委託先である住友電工システムズ(株)及び(有)シーフィックソフトウェア並びに関係機関の関係者の御理解と御協力を頂いており、ここに深く感謝申し上げますとともに、今後の地域保険医療福祉システムの更なる発展を希望するものである。

（国民健康保険補助事業）

昭和62年度	受診者健康管理データバンクシステム事業
63年度	健康管理データバンクシステム事業
平成2年度	総合データバンクシステム事業（5ヵ年度補助）
	現在に至る。

病院経営の基盤安定には
最適なシステムが必須です。
だから、アクセル

アクセル効果①

病院全体の業務効率を高め、患者の待ち時間短縮を実現(サービス向上)

病院の各部門で発生した情報の加工・追加・変換・時間管理などを、利用する人のニーズに合わせて自動的に行ないます。これにより業務効率を高めるとともに、待ち時間の短縮をはじめとする患者サービスの向上を実現します。

アクセル効果②

患者情報、経営情報を一元管理(情報の有効利用)

各部門で入力されたすべての情報を蓄積、患者ごとに一元管理し、診断や経営戦略に役立つ各種情報に整理できます。その情報は部門ごとに設計された入力画面でマウスなどパソコン端末のマンマシン機能を使って容易に共同利用できます。

アクセル効果③

システム化の規模に合わせ段階的に導入(経済性)

システム化の規模や部門業務の増加に合わせ、段階的にアクセルパッケージ及び世界標準UNIXコンピュータなどのハードウェアから選択できる分散処理方式です。つまり(最初の一時期に必要な投資額の削減)が図られ、システムの効率的運用を実現します。

アクセル効果④

ドクターの診療判断や研究のサポート役として(機能の充実)

一元管理された患者情報から基本属性や診療歴の即時表示、検査結果変動分析などの診療支援システムのほか統計解析などの研究支援システムが充実しています。

アクセル効果⑤

ネットワーク型システム(発展性)

さまざまな検査機器やすでに導入済みのコンピュータと接続し、異機種間ネットワークを実現。さらに、病院全体のLANを構築し、統合情報ネットワークへと発展できます。

病院総合情報システム

病院経営を強力にサポートします。

ACCEL

ネットワーク型病院/診療所総合情報システム—アクセル

◆ 住友電気工業株式会社

情報通信システム事業部 MEシステム部システム営業課

◆ 住友電工システムズ株式会社

応用システム事業部

(お問合せは、上記両社とも)

東京都港区元赤坂 1-3-12 〒107 ☎(03)3423-5880(代表)
大阪市西区土佐堀1-2-37 幸福ビル11F 〒550 ☎(06)447-7151(代表)
名古屋市東区東桜1-1-6 住友商事ビル 〒461 ☎(052)963-2755(代表)

DTMとは？

- パソコン用M言語総合開発システム
- マルチユーザーマルチタスクの分散データベースシステム
- 驚異の実行速度、LAN対応！

あなたに、何故 DTM をお薦めするのでしょうか？



アプリケーションソフトウェア、ハードウェア資源を将来に渡って無駄にしません
スクリーンエディター、動的ブレイク点、スタック表示、自動エラーロギングを含む総合開発環境です
LANの性能を引き出すサーバー・クライアント・コンピューティングです
パソコンシステムなので、他のOS環境よりもシステムの取り扱いが簡単です



D A T A T R E E

A Division of InterSystems Corporation

総輸入元

◆ パソコン用高性能M言語処理系
DataTree M言語 (DTM-PC, DT-MAX)

◆ オブション
DT Network
DT Windows (MS Windows 3.0以上に対応)
M/SQL (ODBC対応)

◆ 対応機種
DOS/V 対応パソコン
各社AX規格パソコン
NEC PC9801シリーズ

日本ダイナシステム (株)

Japan DynaSystems Inc.

〒460 名古屋市中区新栄2-1-9 雲竜ビル東館504
Tel.052-242-5441 Fax.052-242-5984

NIFTYServe GGG03270 JDS

いつも新しい扉をオープンするのは DSMです

オープンな稼働環境

- OpenVMS
- DEC OSF/1 (UNIX)
- Windows NT (対応予定)
- Windows 4.0 (対応予定)

オープンクライアント・サーバ環境に対応

- M Windows APIによるMotif及びWin32
プログラミングインターフェース
- DSM API for WindowsによるWindows
パソコンからのアクセス

オープンなデータベースアクセス

- SQL/DSMを利用したSQLでの
Mデータベースアクセス
- Microsoft ODBCサポート

digital

日本DEC

日本 デジタルイクイップメント株式会社

本社/東京都杉並区上荻1-2-1 〒167 TEL03-5349-7111

DSMに関するお問い合わせは、TEL0424-69-6630またはTEL06-222-9211まで
※記載の社名、製品名は各社の商標および登録商標です。

●教務事務作業を強力に支援する

総合成績管理システム

●新しい情報戦略のキーステーションを創造する！

図書・資料管理システム TNLIB Ver.2

当社開発システム

企業

- ・人材管理
- ・資金計算
- ・図書管理
- ・備品管理
- ・顧客名簿管理
- ・マイコン開発
- ・O.A.制御

図書館 資料館 博物館 美術館

- ・図書管理
- ・資料管理
- ・備品管理
- ・名簿管理
- ・収蔵管理
- ・会員管理
- ・イベント資料管理

専門学校 短期大学 大学

- ・入試処理
- ・成績処理
- ・履修管理
- ・学習指導
- ・図書管理
- ・備品管理
- ・進路指導

小学校 中学校 高等学校

- ・入試処理
- ・成績処理
- ・図書管理
- ・備品管理
- ・PTA名簿管理
- ・教材作成支援
- ・進路指導
- ・学習指導

市役所 病院 公共団体

- ・住民管理
- ・施設情報管理
- ・資料管理
- ・情報公開
- ・カルテ管理
- ・薬品管理

株式会社 **高崎共同計算センター**

ヒューマンサポート事業本部 〒370 高崎市米町4-11 第2原ビル 0273(27)2021(代) FAX.0273(27)3483

九段センター 〒102 千代田区飯田橋1-12-15 福岡第四ビル 03(3221)0901(代) FAX.03(3221)0903

多摩センター 〒206 多摩市鶴牧1-4-17 いずみビル3F 0423(73)6855(代) FAX.0423(73)6840